



TECHNICKÁ UNIVERZITA V LIBERCI  
Fakulta mechatroniky, informatiky  
a mezioborových studií



# Systém řízení laboratorní filtrační jednotky

## Bakalářská práce

**Studijní program:** B2612 – Elektrotechnika a informatika

**Studijní obor:** 1802R022 – Informatika a logistika

**Autor práce:** Michal Klaban

**Vedoucí práce:** Ing. Lenka Kretschmerová, Ph.D., ING. PAED. IGIP





TECHNICAL UNIVERSITY OF LIBEREC  
Faculty of Mechatronics, Informatics  
and Interdisciplinary Studies ■

# Control system for lab-scale filtration unit

## Bachelor thesis

**Study programme:** B2612 – Electrical Engineering and Informatics

**Study branch:** 1802R022 – Informatics and Logistics

**Author:** Michal Klaban

**Supervisor:** Ing. Lenka Kretschmerová, Ph.D., ING. PAED. IGIP



## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

<sup>K</sup>(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Michal Klaban  
Osobní číslo: M12000324  
Studijní program: B2612 Elektrotechnika a informatika  
Studijní obor: Informatika a logistika  
Název tématu: Systém řízení laboratorní filtrační jednotky  
Zadávací katedra: Ústav mechatroniky a technické informatiky


### Z á s a d y   p r o   v y p r a c o v á n í :

1. Seznamte se s funkcí cirkulačního obvodu a proveďte rešerši navrženého systému řízení laboratorní filtrační jednotky.
2. Seznamte se s procesorem Arduino a jeho řízením pomocí LabVIEW.
3. Realizujte řízení laboratorní filtrační jednotky v programovém prostředí LabVIEW podle požadavků zadavatele.
4. K vytvořenému řízení laboratorní filtrační jednotky vytvořte návod na obsluhu.


Rozsah grafických prací: dle potřeby dokumentace  
Rozsah pracovní zprávy: cca 30–40 stran  
Forma zpracování bakalářské práce: tištěná/elektronická  
Seznam odborné literatury:

- [1] Kretschmerová, L., Vlach, J.,: Programování v LabVIEW v příkladech, skriptum Liberec, TUL 2014, 144 s., ISBN 978-80-7494-167-2
- [2] Ďaďo, Stanislav. Senzory a měřicí obvody. 2. vyd. Praha: Vydavatelství ČVUT, 1999, 315 s. ISBN 80-010-2057-6

Vedoucí bakalářské práce: Ing. Lenka Kretschmerová, Ph.D., ING. PAED.  
IGIP  
Ústav mechatroniky a technické informatiky  
Konzultant bakalářské práce: Mgr. Ing. Lukáš Dvořák, Ph.D.  
Ústav nových technologií a aplikované informatiky  
Datum zadání bakalářské práce: 10. října 2015  
Termín odevzdání bakalářské práce: 16. května 2016

  
prof. Ing. Václav Kopecký, CSc.  
děkan



  
doc. Ing. Milan Kolář, CSc.  
vedoucí ústavu

V Liberci dne 10. října 2015

## Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Současně čestně prohlašuji, že tištěná verze práce se shoduje s elektronickou verzí, vloženou do IS STAG.

Datum: 11.5.2016

Podpis: 

## **Poděkování**

Rád bych na tomto místě poděkoval vedoucí bakalářské práce paní doktorce Lence Kretschmerové za motivaci, podporu a veškeré vynaložené úsilí k realizaci celého projektu. Dále bych pak rád poděkoval panu doktoru Lukáši Dvořákovi za ochotu odpovídat na veškeré dotazy a pomoc při testování systému filtrační jednotky.

Dále bych chtěl poděkovat firmě ČEZ, a. s. za finanční podporu mé bakalářské práce, díky které byl zakoupen řídicí systém Arduino a veškeré potřebné elektrotechnické součástky pro realizaci kompletního řízení.

## **Absrakt**

Bakalářská práce se zabývá vytvořením automatického systému řízení laboratorní filtrační jednotky. Řídicí systém je realizován pomocí platformy Arduino s programováním ve vývojovém prostředí LabView.

Práce zahrnuje popis původního systému řízení a přehled navrhnutého automatické řízení, kterým se zabývá bakalářský projekt „Návrh automatického řízení cirkulačního obvodu filtrace kalových vod“ od pana Lukáše Pelce. Další část se zabývá pojednáním o platformě Arduino a možnostech jeho programování. Dále pak připojení jednotlivých komponent k desce Arduino, které jsou potřebné pro realizaci systému řízení, a jejich ovládání pomocí vývojového prostředí LabView. V další části je popsáno, jak byly fyzické komponenty naistalovány na filtrační zařízení a jelikož se jedná o systém pracující s kapalinou, je zde popsáno zabudování elektronických součástek do ochranného obalu. Závěrečná část práce se zabývá naprogramováním konečného regulačního obvodu ve vývojovém prostředí LabView.

Výsledkem práce je funkční, samostatně řízený systém filtrační jednotky, který zaznamenává výsledek filtrace a umožňuje uživateli ovládání systému od plně automatického režimu po plně manuální režim.

### **Klíčová slova:**

Arduino, LabView, automatizace, filtrace, Eagle.

## **Abstract**

The aim of this thesis is creating automatic control system for laboratory filtration unit. The control system is implemented in development environment LabView for Arduino platform.

The thesis includes a description of the original control system and an overview of the proposed automatic control which addresses undergraduate project "Design of the automatic control of a circulation circuit filter of sludge water" by Mr. Lukáš Pelc. Another part is focused on discussion about Arduino platform and the possibilities of its programming. Next part is connection of various components, which are necessary for the implementation of the control system, to the Arduino platform and their control using LabView. The next section describes how the physical components were installed on the filter device and since it is a system that operates with a liquid there is also a description of incorporation of electronic components in a protective case. The final part is focused on the programming of the final regulatory circuit in LabView development environment.

The result of this undergraduate project is a functional, self-controlled system of a filter unit which records the result of filtering and allows the user to control the system from fully automatic mode to full manual mode.

### **Key words:**

Arduino, LabView, automatization, filtration, Eagle.



# Obsah

1	Úvod.....	12
1.1	Úvod do automatizace.....	12
1.2	Cíl práce .....	12
2	Původní systém filtrace a návrh nového .....	13
2.1	Filtrační jednotka .....	13
2.2	Původní systém filtrace .....	14
2.3	Navržený automatizační obvod.....	15
2.4	Navržené komponenty pro řízení .....	15
2.4.1	Ventily .....	15
2.4.2	Průtokoměry.....	16
2.4.3	Tlakovoměry.....	16
3	Arduino .....	17
3.1	Technické informace o deskách Arduino.....	17
3.2	Zvolené Arduino Nano 3.0.....	17
4	Programování Arduino .....	20
4.1	Připojení Arduino desky k počítači.....	20
4.2	Programování Arduino v Arduino IDE.....	22
4.3	Programování Arduino v LabView .....	24
4.3.1	Stahování balíčku pro Arduino .....	24
4.3.2	Zahájení programování Arduina v Labview .....	27
4.3.3	Základy programování Arduina v LabView .....	29
5	Připojení komponent k Arduino .....	31
5.1	Třícestné ventily.....	31
5.2	Snímače průtoku.....	33
5.3	Snímače tlaku .....	35
5.4	Ovládání čerpadla.....	36

5.5	Ventil se servopohonem .....	37
6	Fyzická kompletace filtračního systému.....	39
6.1	Instalace komponent na filtrační zařízení .....	39
6.1.1	Instalace třicestných ventilů.....	39
6.1.2	Instalace snímačů průtoků .....	40
6.1.3	Instalace snímačů tlaku .....	40
6.1.4	Instalace ventilu se servopohonem .....	41
6.2	Návrh plošných spojů – Eagle.....	42
6.3	Ochranný obal .....	43
7	Vizuální podoba programu .....	44
7.1	Ovládací panel programu .....	44
7.2	Zobrazení měřených hodnot.....	44
8	Program pro ovládání filtrační jednotky .....	46
8.1	První část programu: inicializace Arduina .....	46
8.2	Druhá část programu: inicializace hodnot v Arduinu .....	47
8.3	Třetí část programu: získání času zahájení měření .....	47
8.4	Čtvrtá část programu: průběh měření.....	48
8.4.1	Zobrazování grafů, nastavování servopohonu a čtení počtu otáček .....	48
8.4.2	Ukončení měření při Erroru a řízení podle tlaku nebo průtoků .....	48
8.4.3	Měření tlaku .....	48
8.4.4	Změna hodnot uživatelem.....	49
8.4.5	Uložení hodnot do pole a výpočet permeability .....	49
8.5	Pátá část programu: ukončení .....	49
9	Tvorba manuálu .....	50
10	Závěr .....	52
	Seznam použité literatury .....	53
	Seznam příloh .....	55

## Seznam použitých zkratek a značek

IDE	- Integrated development enviroment	- Integrované vývojové prostředí
SRAM	- Static Random Access Memory	- Statická paměť
PROM	- Programmable Read Only Memory	- Programovatelná paměť
EEPROM	- Electrically Erasable PROM	- Elektricky mazatelná paměť
USB	- Universal Serial Bus	- Univerzální sériová komunikace
COM port	- Communication port	- Komunikační rozhraní
LIFA	- LabView Interface For Arduino	- Rozhraní LabView pro Ardino
PWM	- Pulse-width modulation	- Pulzně šířková modulace
FTDI	- Future Technology Devices International	- Firma zaměřená na USB
VSS / VEE	- Záporný napájecí potenciál	/ Zem
VDD / VCC	- Kladný napájecí potenciál	
.Vi	- Koncovka programů vytvořená v LabView	

Značka		Jednotka	
U	- Napětí	- V	- Volt
f	- Frekvence	- Hz	- Hertz
p	- Tlak	- Pa / Bar	- Pascal / Bar
Q	- Průtok	- l/h	- Litr za hodinu
P <sub>s</sub>	- Výkon	- W	- Watt
S	- Plocha	- m <sup>2</sup>	- Metr čtvereční
t	- Teplota	- °C	- Stupeň celsia
R	- Odpor	- Ω	- Ohm
C	- Elektrická kapacita	- F	- Farad

## Seznam obrázků

Obrázek 2.1: Původní vzhled filtrační jednotky bez cely .....	13
Obrázek 2.2: Původní ovládání filtrační jednotky .....	14
Obrázek 2.3: Návrh automatizačního obvodu [4].....	15
Obrázek 3.1: Arduino Nano s FTDI [12].....	18
Obrázek 3.2: Aurduino Nano - rozmístění pinů [12].....	19
Obrázek 4.1: Zobrazení COM portu ve Správci zařízení ve Windows 7 .....	21
Obrázek 4.2: Vlastnoti COM portu ve Správci zařízení ve Windows 7.....	21
Obrázek 4.3: Pokročilé vlastnosti COM portu ve Správci zařízení ve Windows 7.....	22
Obrázek 4.4: Spuštěná aplikace Arduino IDE .....	23
Obrázek 4.5: Základní okno JKI VI Package manager .....	25
Obrázek 4.6: JKI VI Package manager po zvolení balíčku .....	26
Obrázek 4.7: Nainstalovaný nástroj pro Arduino v programu LabView.....	27
Obrázek 4.8: Otevření a kompilace LIFA Base.ino.....	28
Obrázek 5.1: Průtoky kapaliny třícestným ventilem – a) klidový stav; b) pod napětím	31
Obrázek 5.2: Propojení Arduina a dvou relé v relé modulu .....	32
Obrázek 5.3: Propojení Arduina a průtokoměru.....	34
Obrázek 5.4: Schéma jednoduchého odporového děliče .....	35
Obrázek 5.5: Propojení Arduina a tlakoměru .....	36
Obrázek 5.6: Schéma zapojení součástek v program Eagle .....	38
Obrázek 6.1: Uchycení komponent na filtračním zařízení .....	39
Obrázek 6.2: Fyzické uchycení třícestných ventilů .....	40
Obrázek 6.3: Umístění průtokoměrů .....	40
Obrázek 6.4: Fyzický vzhled snímačů se speciálním uchycením.....	41
Obrázek 6.5: Fyzické připojení servopohonu .....	41
Obrázek 6.6: Plošný spoj pro PWM regulaci .....	42
Obrázek 6.7: Plošný spoj pro čítač, odporový dělič a rozvody .....	43

Obrázek 6.8: Ochranný obal .....	43
Obrázek 7.1: Ovládací panel programu filtrační jednotky .....	44
Obrázek 7.2: Zobrazovací panel hodnot programu filtrační jednotky .....	45
Obrázek 8.1: Graf základních bloků programu filtrační jednotky .....	46
Obrázek 9.1: Titulní stránka manuálu .....	51

## Seznam tabulek

Tabulka 1: Technická data filtrační jednotky [3] .....	14
Tabulka 2: Propojení konektorů relé modulu a Arduina nano .....	32
Tabulka 3: Parametry průtokoměru [15] .....	33
Tabulka 4: Propojení konektorů průtokoměru a Arduina nano .....	34
Tabulka 5: Propojení konektorů tlakoměru a Arduina nano .....	35
Tabulka 6: Součástky pro zesílení PWM signálu .....	37

## Seznam zdrojových kódů

Zdrojový kód 1: Základní prvky Arduina pro programování v LabView .....	29
Zdrojový kód 2: Ovládání module relé v programu LabView .....	32
Zdrojový kód 3: Čtení počtu otáček průtokoměru v programu LabView .....	34
Zdrojový kód 4: Čtení hodnoty z tlakoměru v programu LabView .....	36
Zdrojový kód 5: Ovládání servopohonu v programu LabView .....	38
Zdrojový kód 6: Inicializace Arduina .....	47
Zdrojový kód 7: Třetí část programu: získání času zahájení .....	47

# 1 Úvod

Oblasti automatizace je v současné době jeden z hlavních směrů technického vývoje založený na zavádění a užívání samočinných zařízení bez obsluhy.

## 1.1 Úvod do automatizace

Automatizace se dá rozdělit do dvou základních podskupin, a to automatizace fyzické práce a automatizace duševní činnosti. Automatizace duševní činnosti v dnešní době je zjednodušeně řečeno počítač. Od elektrického stroje pro sčítání obyvatelstva v roce 1896 s názvem Hollerith je vývoj počítačů velice svižný. Mezi důležité historické počítače patří například reléový počítač Mark I a elektronkový počítač Eniac. V roce 1947 byl objeven tranzistor, díky čemuž se rozmohl vývoj výpočetní techniky a začaly se objevovat i domácí počítače a mnoho jiné výpočetní techniky, která nám aktuálně pomáhá v každodenním životě. [1]

Automatizace fyzické práce se postupně vyvíjí již od starověku s prvními pákami, kladkami, koly, párou přes středověk a rozvoj různých mechanismů jako jsou čerpadla, mlýny, hodiny a další vynálezy ulehčující fyzickou práci. V průmyslové revoluci novověku se dá vyzdvihnout například první regulace pomocí Wattova odstředivého regulátoru z roku 1775 nebo Jacquardův tkalcovský stav s programovacím děrovaným páskem z roku 1804. Komplexnější automatizaci pak velice přispěl první montážní výrobní linkou hromadné výroby v roce 1913 Henry Ford. [1] [2]

Automatizace v sobě zahrnuje mnoho pozitivních aspektů, jako například možnost snížení výrobních nákladů, zvýšení produktivity a kvality, zkrácení dob vývoje a výroby, pružná reakce na požadavky trhu, zvyšování pohodlí člověka, zvyšování požadavků na ekologii.

## 1.2 Cíl práce

Cílem této bakalářské práce bylo vytvoření automatického systému řízení laboratorní filtrační jednotky, která slouží k měření membrán tak aby při dlouhodobém měření nebyla potřeba fyzická přítomnost člověka.

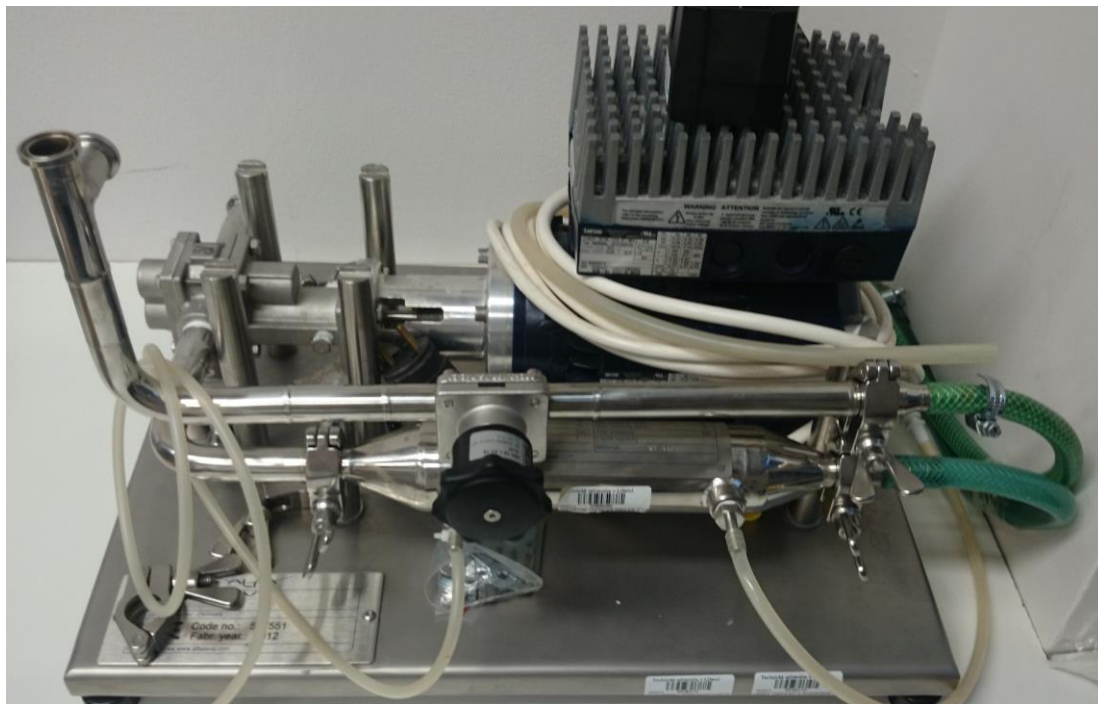
Cíl bakalářské práce se rozdělil do tří hlavních úkolů. Prvním úkolem je ovládání navrhnutých měřících a řídicích prvků pomocí Arduina. Těmito prvky jsou tlakoměry, průtokoměry, ventily a ovládání čerpadla. Druhý úkol se zabývá fyzickou implementací měřících a řídicích prvků na samotnou filtrační jednotku. Třetí úkol řeší celkové naprogramování filtračního systému a uživatelské rozhraní, u kterého je potřeba dodržet možnost výběru průběhu filtrace a uložení naměřených hodnot v celém průběhu filtrace.

Výběr průběhu filtrace:

- a) plně manuální režim;
- b) s konstantním průtokem;
- c) s konstantním tlakem.

## 2 Původní systém filtrace a návrh nového

Systém filtrace byl navrhován na cross-flow filtrační jednotce s názvem Alfa Laval labUnit M10, která je zobrazena na obrázku 2.1 a je určena pro mikrofiltrační a ultrafiltrační membrány.



Obrázek 2.1: Původní vzhled filtrační jednotky bez cely

### 2.1 Filtrační jednotka

Filtrační jednotka disponuje čtyřmi membránovými celami, které jsou spojeny v párech. Filtrační jednotka se skládá ze systému trubek pro vstup a výstup retentátu. Retentát je termín používaný v membránové technologii pro tekutinu, která je zachována v procesu filtrace. Pro výstup a vstup retentátu jsou určeny 14mm hadičky. Kapalina, která proteče filtrem, se nazývá permeát. Pro odtok permeátu jsou určeny 4mm hadičky.

Dalšími součástmi jednotky je elektrický zdroj (3 x 400 V, 50/60Hz), dva nanometry (0-700 kPa), pumpa ECO (4-7 l/h, 7 bar, 0,37 kW), tepelný výměník, termostat Lauda ECO Silver a čerpadlo Lenze (0,66 kW, 87 Hz). Další technická data jsou uvedena v tabulce 1. [3]

Tabulka 1: Technická data filtrační jednotky [3]

Parametr	Hodnota/jednotka
Počet membrán	4
Plocha jedné membrány	0,0084 m <sup>2</sup>
Cross-flow rychlost	3-10 l/min
Maximální povolená teplota*	80 °C
Maximální povolený tlak*	700 kPa

\* Hodnota se může lišit na použitém typu membrán

## 2.2 Původní systém filtrace

Původní systém byl navržen pouze pro manuální ovládání tak, že bylo potřeba filtrační zařízení neustále hlídat. Bylo nutné hlídat tlak v systému, díky čemuž bylo zřetelné zanesení vloženého filtru. Dále pak bylo zapotřebí hlídat čistou kapalinu, zda již nádoba určená pro přefiltrovanou kapalinu nebyla plná. Z důvodu neustálé potřeby kontrolovat byla nejvhodnějším řešením automatizace tohoto systému.

Původní ovládání, které je zobrazeno na obrázku 2.2, bylo realizováno pomocí otočné hlavice, jež udávala rychlost čerpadla a možnosti přepínání tří stavů. První stav byl chod čerpadla v před, nula znázorňovala vypnuté čerpadlo a druhý stav byl zpětný chod čerpadla.

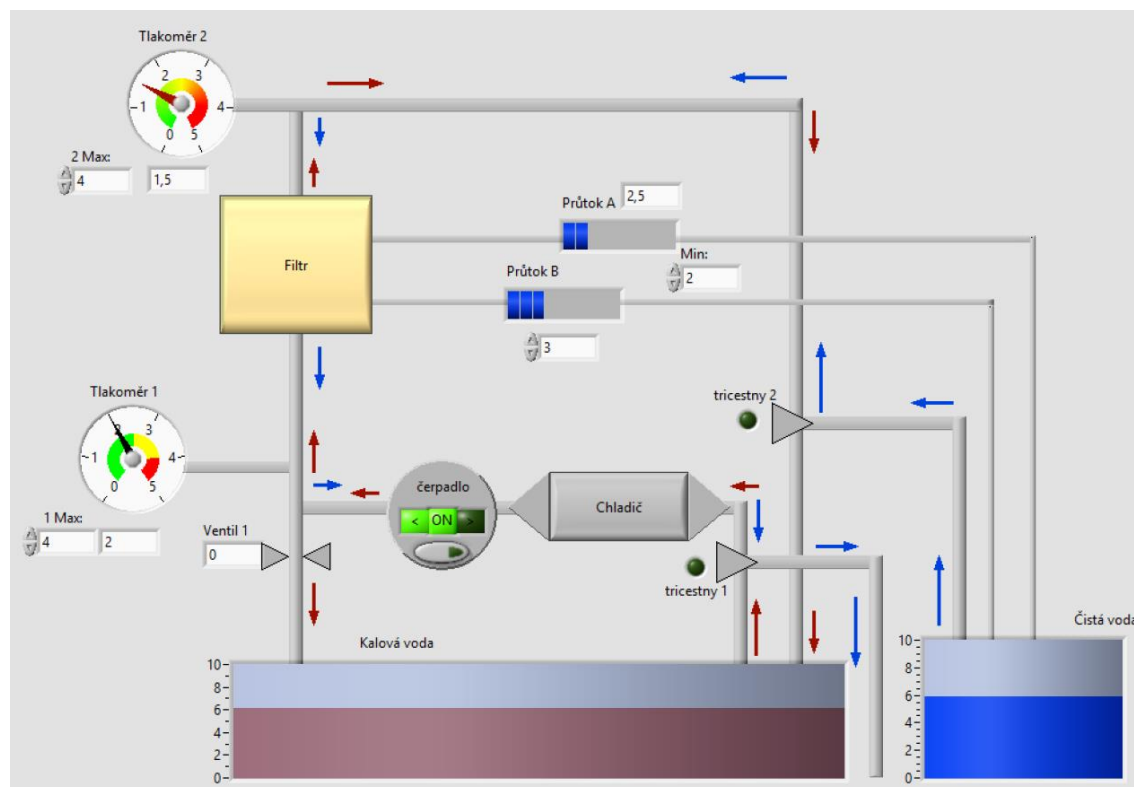


Obrázek 2.2: Původní ovládání filtrační jednotky



## 2.3 Navržený automatizační obvod

Filtrační obvod byl navržen tak, aby byla možnost regulovat soustavu na konstantní tlak nebo průtok. Navíc je zde možnost vyčistit filtr pomocí čisté vody nastavením zpětnému chodu. Tento navržený filtrační obvod je zobrazen na obrázku 2.3.



Obrázek 2.3: Návrh automatizačního obvodu [4]

## 2.4 Navržené komponenty pro řízení

Komponenty byly vybrány tak, aby filtrační jednotka byla zcela automatická.

### 2.4.1 Ventily

Pro možnost oběhu dvou různých kapalin byly zvoleny dva třicestné ventily. Díky těmto ventilům je možnost střídát oběh s čistou vodou pro čištění filtru a oběh s kalovou vodou pro filtrování. Výsledná kapalina po filtrování se nazývá permeát.

V původním návrhu, viz obrázek 2.3, teče permeát do čisté vody, kterou se následně dá čistit filtr. Tento návrh se dá změnit podle toho, do jaké nádoby se vstupní a výstupní hadice vloží.

Pro výběr konkrétní kapaliny v oběhu jsou zvoleny třicestné zónové ventily s vratnou pružinou VZP 315-230-1P. Tyto ventily se ovládají mocí napájení, které je zapnuté nebo vypnuté. [5]

Třetí ventil je jednocestný kulový Belimo R2015-P63-S1 pro zajištění vhodného tlaku v oběhu ovládaný přes servopohon Belimo-LR24A-SROV, který je ovládaný pomocí PWM signálu. [6] [7]

### **2.4.2 Průtokoměry**

Průtokoměry slouží pro výstupní hodnocení z filtrační jednotky.

Byly zvoleny průtokoměry B.I.O-TECH e.K. FCH-m-POM-LC, které by měly být dostatečné pro zaznamenávání výstupní kapaliny. Výstup z průtokoměru je signál s určitým počtem pulsů na litr. [8]

### **2.4.3 Tlakoměry**

Původní budíkové tlakoměry byly nahrazeny snímači relativního tlaku D2410. Tyto snímače udávají velikost tlaku přímo úměrné velikosti výstupního napětí ze snímače. [9]

### 3 Arduino

Jedná se o otevřenou softwarovou platformu pro návrh a vývoj elektronických programovatelných zařízení. Tato platforma nabízí možnost konstrukce prototypů od hardwarové části až po jeho programování s cílem co největšího zjednodušení pro možnost vývoje širokou veřejností. Díky otevřenosti této platformy je možné si upravovat její části podle vlastních potřeb s podporou rozsáhlé komunity, která se vývoji na této platformě zabývá. Technologie Arduino umožňuje postavit si hardware plnící různé funkce bez větších investic. Například je možné sestavit menší server, zobrazování pomocí LED, segmentových displejů, nebo i komplexnějších LCD displejů, regulátor teploty, bezdrátové síťové prvky, výkonové spínací prvky, ultrazvukové senzory, různé řídicí jednotky pro roboty, nebo jiné automaty. S různými požadavky pomáhají i vývojové kity, které jsou nabízeny v různých variantách a nabízejí základní prvky pro zahájení vývoje vlastního zařízení. [10]

#### 3.1 Technické informace o deskách Arduino

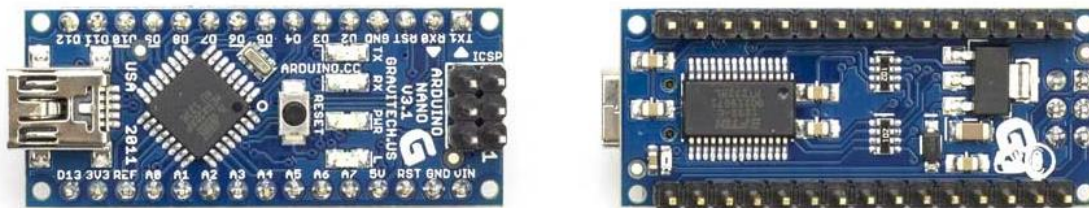
K deskám Arduino lze připojit jakékoliv periferní hardware od malých až po distribuované systémy. Díky této možnosti jsou kladeny různé nároky typ mikrokontroléru. Všechna zařízení Arduino jsou vybaveny mikrokontrolérem ATmega. Patří sem například ATmega8, ATmega168, ATmega328 a další. Tyto čipy se liší velikostí paměti RAM a FLASH. Každá deska může mít jiný mikrokontroler, popřípadě i výměnný. [11]

Desky Arduino jsou dále vybaveny dalšími prvky, například různé počty vstupní a výstupních analogových, nebo digitálních portů, analogově-digitálními převodníky, PWM výstupy, stabilizátory napájení a další. Pro sériové zapojení lze využít porty RS-232 zajišťující komunikaci mezi dvěma hardwary. Jednou z nejdůležitějších vlastností je možnost využití sériové komunikace přes USB port a tím propojení Arduino přes nejdostupnější port s vlastním počítačem nebo jiným hardwarem. Tuto komunikaci zajišťují obvody od FTDI nebo novější ATmega16U2. Díky těmto obvodům může počítač vidět Arduino jako virtuální COM port přes USB. Pomocí USB portu se dají desky Arduino i napájet. [11]

Díky otevřenosti této platformy se objevují klony součástí Arduino. Největší výhodou těchto klonů je cena. Ta se však projevuje na zpracování a možnosti horší kompatibility.

#### 3.2 Zvolené Arduino Nano 3.0

Pro účely této bakalářské práce byla vybrána deska Arduino Nano 3.0 zobrazena na obrázku 3.1. Tato deska byla vybrána jako vyhovující díky její nižší ceně, velikosti a dostačujícímu počtu pinů pro ovládání nebo čtení každého z potřebných prvků pro realizaci automatické filtrační jednotky.



Obrázek 3.1: Arduino Nano s FTDI [12]

Tato deska je osazena osmibitovým mikrokontrolérem Atmel AVR ATmega328. Vývojová deska má 14 digitálních vstupně-výstupních pinů, z nichž lze 6 použít jako výstupy PWM. Dále má 8 analogových vstupů, USB konektor typu Mini-B a tlačítko restartu. Rozmístění vstupů a výstupů z desky je zobrazeno na obrázku 3.2. Deska je vybavena krystalovým oscilátorem pracujícím na frekvenci 16 MHz.

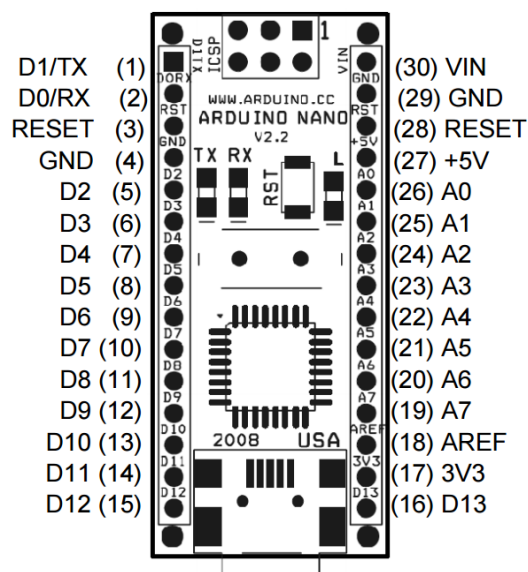
Flash paměť tohoto mikrokontroléru pro uložení programu je 32 KB, kde 2 KB jsou využity pro bootloader neboli zavaděč, který je zodpovědný za zapnutí desky. Dále mikrokontrolér disponuje 2 KB SRAM a 1 KB EEPROM.

Každým digitálním vstupně-výstupním pinem může protékat proud 40 mA. Piny jsou vybaveny pull-up rezistory, které jsou v základním nastavení odpojeny.

Napájení desky může probíhat přes USB port nebo pomocí externího zdroje s napětím 6-20V na pinu 30, popřípadě externím zdrojem s napětím 5 V na pinu 27.

Důležitou součástí této desky je procesor FTDI FT232RL pro řízení sériové komunikace přes USB, který poskytuje virtuální COM port v softwaru počítače.

[12]



Obrázek 3.2: Arduino Nano - rozmístění pinů [12]

Díky velice příznivé ceně byla první deska Arduino Nano objednána jako klon. Tento klon má stejné specifikace, stejné rozložení, stejný osmibitový mikrokontrolér Atmel AVR ATmega328, ovšem odlišný procesor pro řízení sériové komunikace. Kvůli levnějším nákladům je v klonu pro sériovou komunikaci zabudovaný procesor CH340. Tento procesor má odlišnou kompatibilitu a na rozdíl od FTDI, kde většinou ovladače naistalují pomocí operačního systému samy, je potřeba ovladače instalovat zvlášť. Po nainstalování doporučených ovladačů může dojít k problémům se spojením, díky kterým bylo objednáno originální Arduino Nano v3.0 s procesorem pro sériovou komunikaci od FTDI. Tento problém byl v nepřijetí Arduino i přes nainstalované ovladače a různé modifikace těchto ovladačů. Na závěr bylo zjištěno, že některé počítače s Arduinem komunikovaly, v některých byly různé problémy s komunikací a po připojení Arduino do portu USB 3.0 a vyšší se komunikace vůbec nezdařila. Z těchto důvodů byla zvolena originální varianta Arduino.

## 4 Programování Arduino

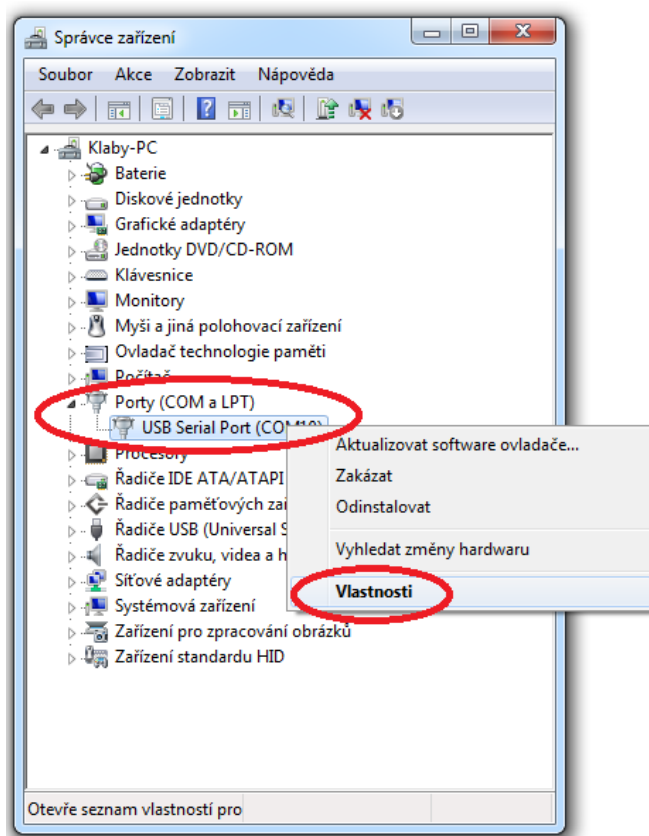
Desky Arduino se dají programovat ve více programovacích jazycích. Jako základní vývojové prostředí přímo navržený pro Arduino je „Arduino IDE“. Jedná se o integrované vývojové prostředí napsané v jazyce Java, díky čemuž je patrné, že se k programování Arduina dá přistupovat i pomocí jazyka Java. Arduino IDE bylo navrženo jako nejjednodušší volba pro kteréhokoli uživatele, který se chce zabývat vývojem aplikací pro Arduino. Toto vývojové prostředí pochází z výukového prostředí Processing, které vychází z jazyka Java a značně jej zjednodušuje. Díky prostředí Processing se dá Arduino programovat ve stejnojmenném jazyce, který se velice podobá jazyku Java. Arduino IDE bylo upraveno pro zjednodušení programování Arduina pomocí přidání určitých funkcí a hlavně byla přidána podpora jazyka Wiring. Další z možností výběru jazyka je programování pomocí C++ ve vývojovém prostředí Visual Studio. [10] [13]

Další z možností výběru jazyka vytvářejí klony Arduina. Například Netduino se dá programovat v jazyce C#, k čemu je i vytvořeno vývojové prostředí „Visual studio IDE for Netduino“. [14]

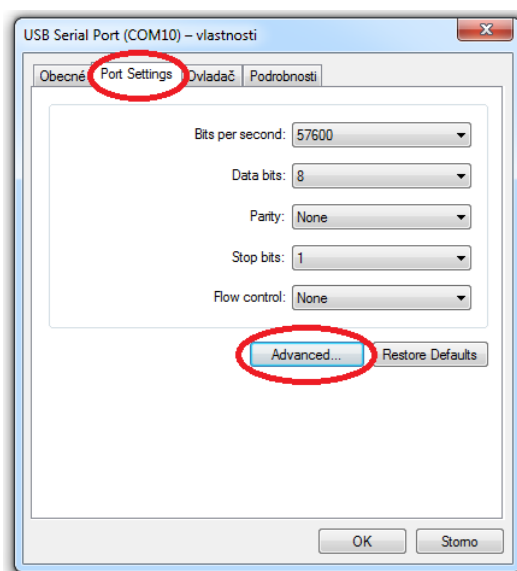
Platforma Arduino je univerzální a rozsáhlá platforma, která se dá programovat v různých jazycích a záleží pouze na preferovaném jazyku. Pro vývoj systému filtrační jednotky bylo zvoleno grafické vývojové prostředí LabView.

### 4.1 Připojení Arduino desky k počítači

Pokud používáme originální desku Arduino s komunikačním čipem od FTDI a operační systém Windows 7 a vyšší, měly by se ovladače pro komunikaci nainstalovat automaticky. Nainstalování ovladačů se dá ověřit ve Windows 7 pomocí „Správce zařízení“. Pokud je ovladač správně nainstalovaný, nalezneme zde řádek „Porty (COM a LPT)“, který po rozbalení obsahuje „USB Serial Port (COM?)“, kde otazník je automaticky přiřazené číslo portu. Takto nainstalovaný ovladač je znázorněn na obrázku 4.1. Pokud je však Arduino nalezeno jako neznámé zařízení, je potřeba na stránkách „<http://www.ftdichip.com/FTDrivers.htm>“ stáhnout příslušný ovladač a ten nainstalovat.



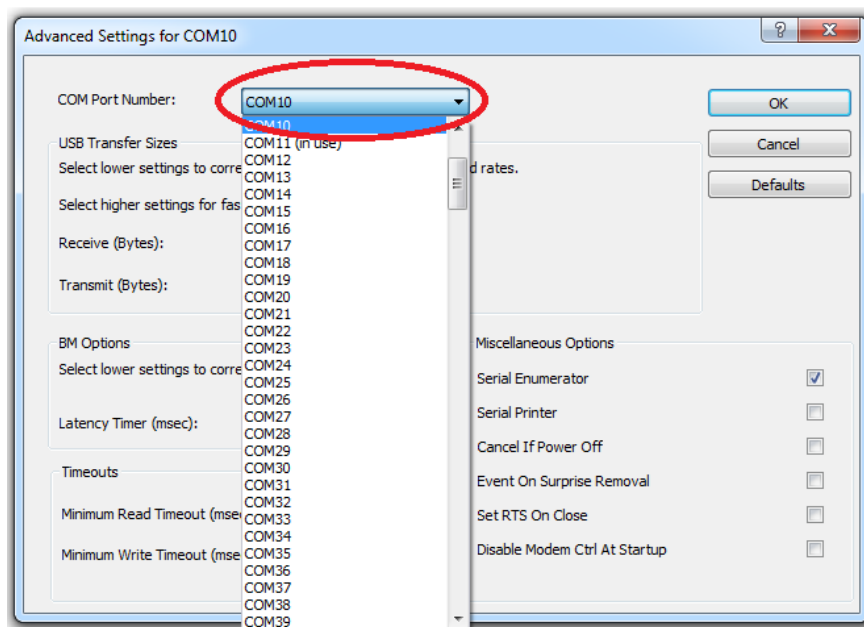
Obrázek 4.1: Zobrazení COM portu ve Správci zařízení ve Windows 7



Obrázek 4.2: Vlastnoti COM portu ve Správci zařízení ve Windows 7

Při využívání více desek Arduino je vhodné si určit ke každé desce vlastní číslo portu. Po připojení desky se objeví správně nainstalované zařízení ve „Správci zařízení Windows“ jako COM port. Na toto zařízení klikneme pravým tlačítkem a zvolíme možnost vlastnosti, viz obrázek 4.1. Dále je potřeba přejít do záložky „Port Settings“

a kliknout na možnost „Advances“, viz obrázek 4.2. Zde je možné zvolit vlastní číslo COM portu pro danou desku. Tato možnost je znázorněna na obrázku 4.3. Pro uložení volby stačí kliknout na tlačítko „OK“ a zbylá okna zavřít.



Obrázek 4.3: Pokročilé vlastnosti COM portu ve Správci zařízení ve Windows 7

Při připojení druhé desky Arduino je vhodné po prvním připojení opět nastavit vlastní číslo COM portu.

## 4.2 Programování Arduino v Arduino IDE

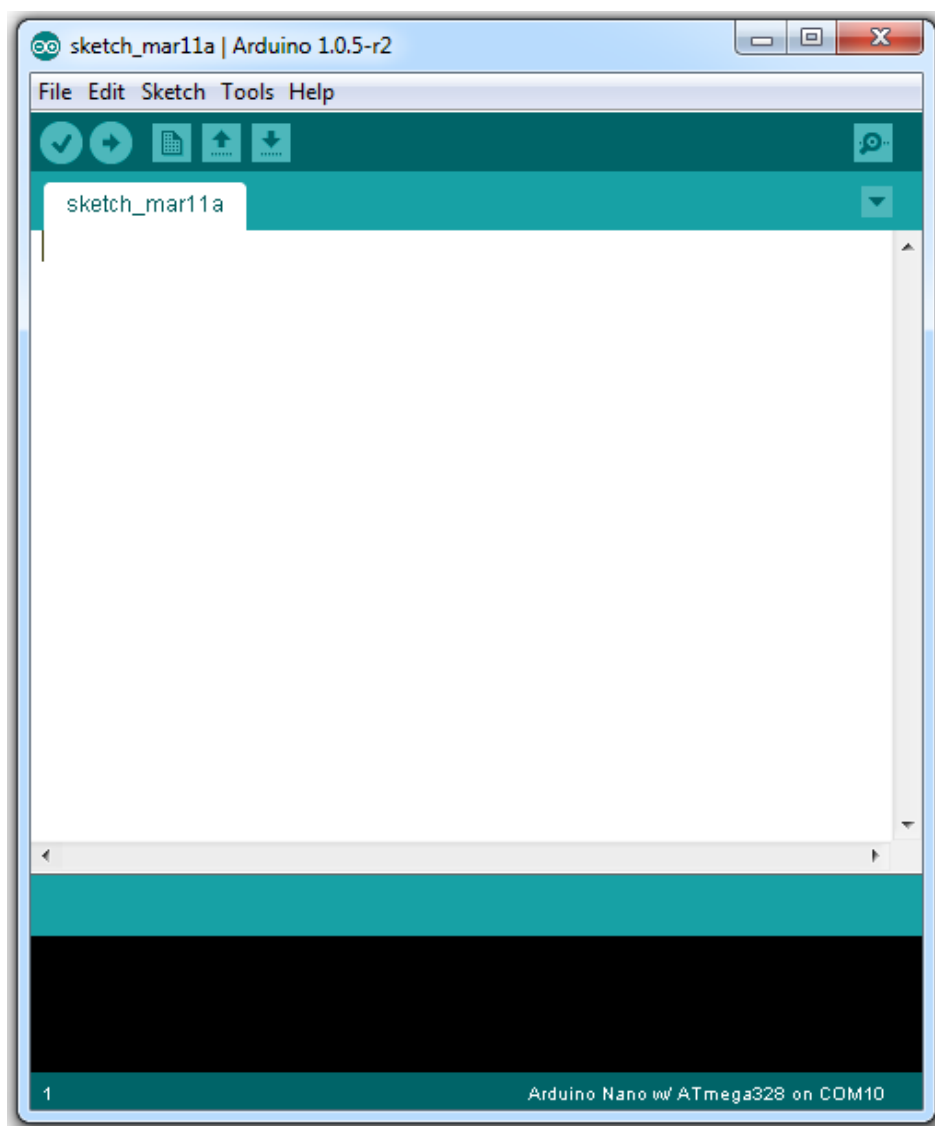
Vývojové prostředí Arduino IDE je možné zprovoznit na většině operačních systémů jako je Windows, Mac OS, nebo Linux kde, zprovoznění a instalace záleží na dané distribuci. Veškeré informace k instalaci vývojového prostředí jsou na oficiálních stránkách Arduino. Arduino IDE je aktuálně dostupné v plně funkční verzi 1.0.5, nebo v testovací verzi 1.5. Pro bezproblémové programování v Arduino IDE je vhodné pracovat ve starší verzi 1.0.5.

Pro účely automatizace filtrační jednotky byly vyzkoušeny verze 1.0.5 i 1.5 na operačním systému Windows 7. Pro spuštění vývojového prostředí je nejvhodnější stáhnout ZIP archiv určený pro systém Windows 7 na oficiálních stránkách Arduino. Po stažení je potřeba archiv ZIP rozbalit. Po rozbalení je již vývojové prostředí Arduino IDE plně funkční a spustí se souborem arduino.exe.

Po spuštění vývojového prostředí se objeví okno aplikace, které je zobrazeno na obrázku 4.4. Pro základní programování v tomto prostředí je důležitá v prvním řádku rozbalovací nabídka „Tools“. Po připojení Arduino k počítači je v této nabídce nutné správně nastavit záložku „Board“ a „Serial port“. Záložka „Board“ obsahuje veškeré podporované desky Arduino a je potřeba vybrat námi používanou desku.



Záložka „Serial port“ po připojení jedné desky Arduino obsahuje většinou pouze jeden možný port, který je třeba zvolit. Pokud však desku přepojíme do jiného USB portu na počítači, může se objevit jiný port a je potřeba tento port v nabídce změnit.



Obrázek 4.4: Spuštěná aplikace Arduino IDE

Druhý řádek ve vývojovém prostředí Arduino IDE obsahuje tlačítka pro zkontrolování kódu, nahrání kódu do Arduino, otevření nového programovacího okna, načtení a uložení kódu.

## 4.3 Programování Arduino v LabView

LabView je grafické vývojové prostředí, ve kterém se zdrojový kód píše v jazyce G, jež reprezentuje grafický jazyk stejně jako u jiných programovacích jazyků strukturovaný text. Například vývojové prostředí Arduino IDE podporuje jazyk Wiring, což je strukturovaný text. Stejně jako pro ostatní programovací jazyky existuje kompilátor pro jazyk G, který vytváří samostatné spustitelné programy. V jazyce G jsou dostupné rychlé programovací základní funkce, tak i kompilované podprogramy pro komunikaci, matematickou analýzu, statistiku a podobně.

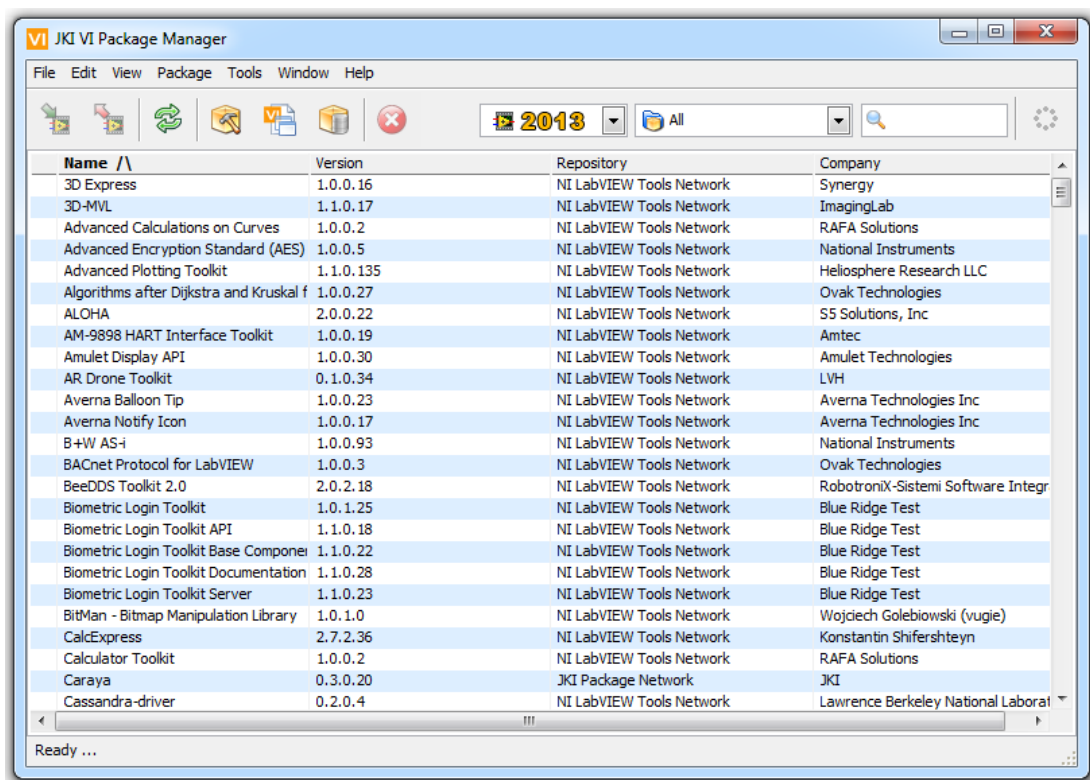
Vývojové prostředí LabView vyvíjí firma National Instruments. LabView obsahuje knihovny funkcí a vývojové nástroje navržené pro získávání dat a ovládání přístrojů.

### 4.3.1 Stažení balíčku pro Arduino

Díky obsáhlé podpoře by bylo nepřehledné, pokud by základní program LabView obsahoval veškeré podporované vývojové nástroje. Pro stažení potřebných nástrojů pro programování obsažených v balíčcích se stahují pomocí doplňku „JKI VI Package Manager“.

Tento doplněk je obsažen v některých instalacích LabView. Pokud je již nainstalován, stačí doplněk v počítači vyhledat a spustit. Pokud doplněk nainstalován není, je poskytnut na adrese „<http://jki.net/vipm>“, kde je potřeba stáhnout „VI Package Manager“.

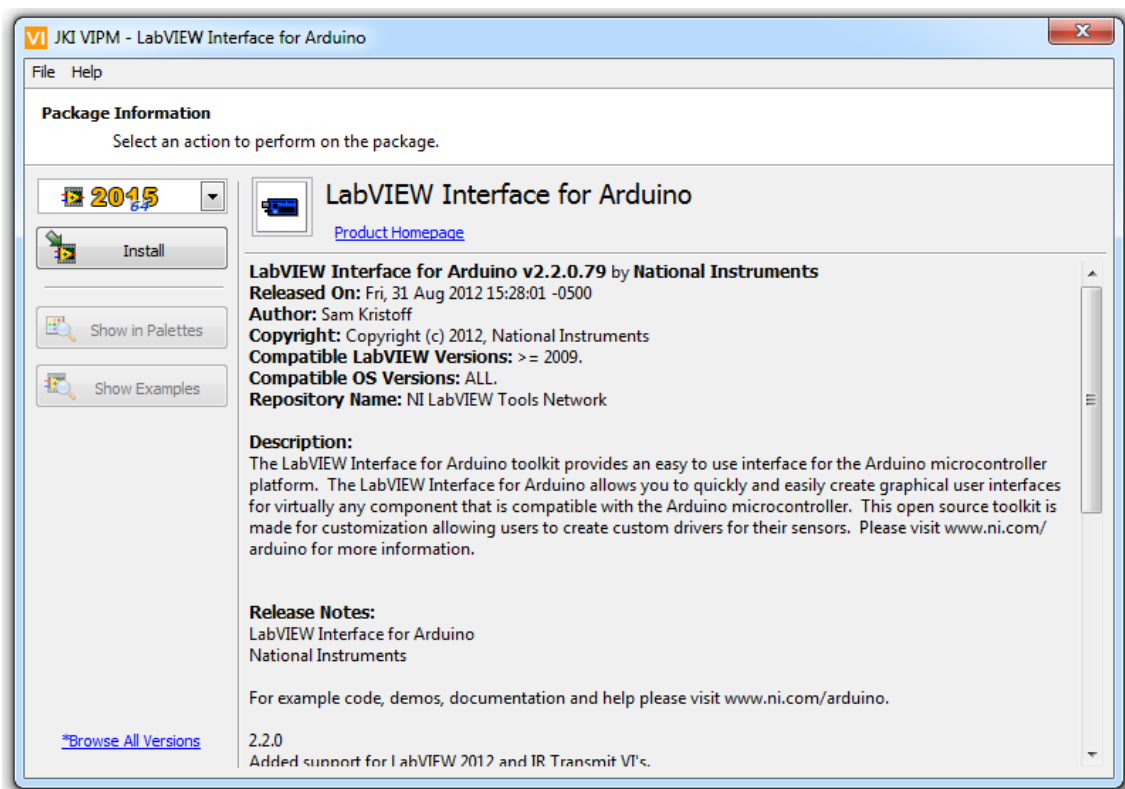
Po nainstalování je možné tento program spustit. Při správném spuštění se zobrazí základní okno, které je zobrazeno na obrázku 4.5. V tomto programu jsou nejdůležitější první dva obrázky, které umožňují instalovat, nebo odinstalovat balíčky z LabView. Dále pak obsahuje zvolení verze LabView, pro které jsou zobrazené nástroje určeny, a vyhledání potřebného balíčku.



Obrázek 4.5: Základní okno JKI VI Package manager

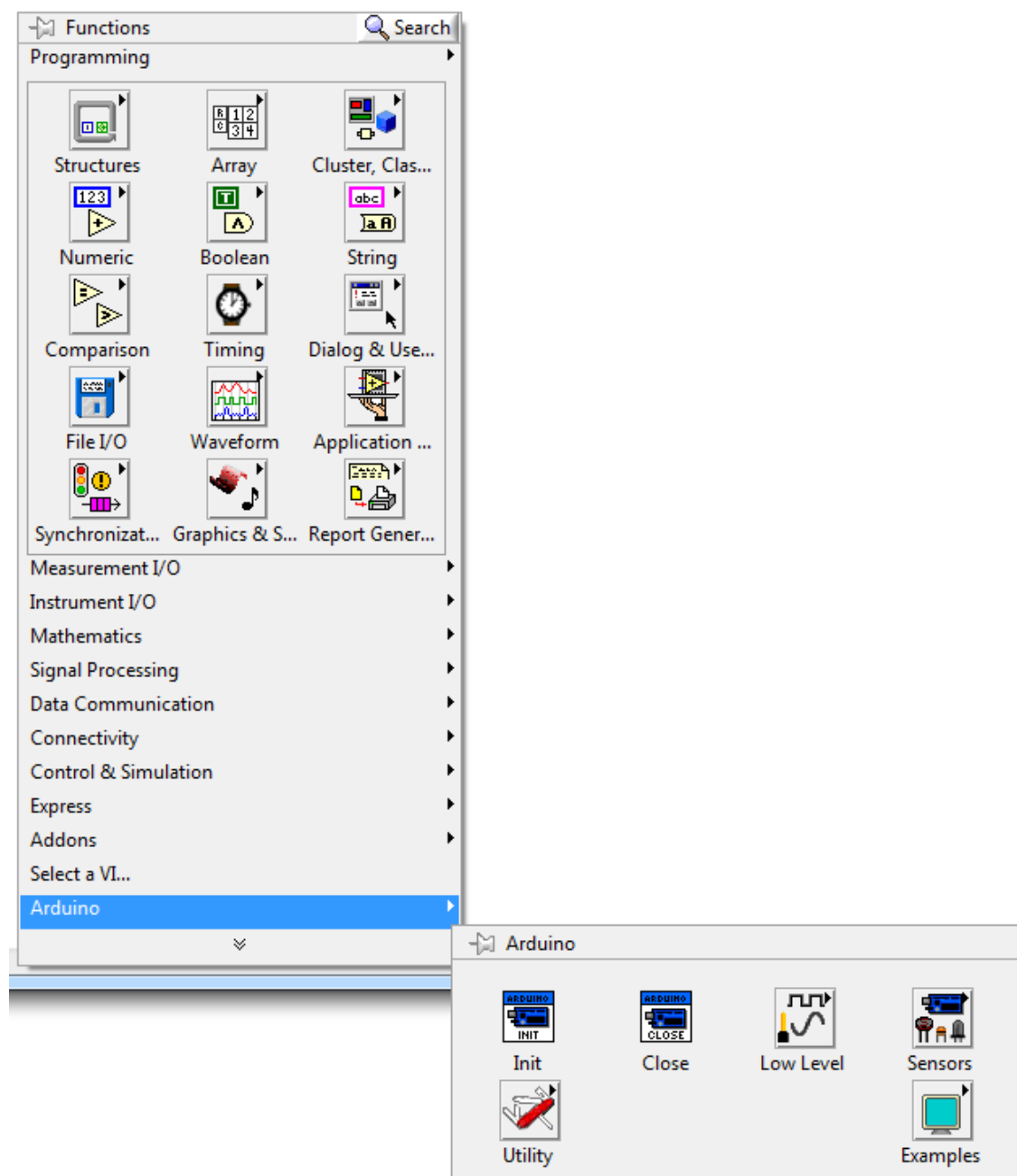
Pro účely využití Arduina je potřeba zvolit vhodnou verzi LabView a vyhledat balíček „LabVIEW Interface for Arduino“. Tento balíček obsahuje nástroje pro jednoduché ovládání desky Arduino přes LabView. Jedná se o otevřenou platformu v rámci celého projektu Arduino, díky čemuž je možné tento balíček využívat bezplatně.

Po vyhledání tohoto balíčku je vhodné tento balíček označit a kliknout na možnost „Instal package(s)“, popřípadě je možné balíček otevřít, kde se objeví okno, které je zobrazeno na obrázku 4.6. V tomto okně jsou zobrazeny informace ohledně otevřeného balíčku. Zde je potřeba zvolit verzi LabView, do které je nutné doplněk nainstalovat a následně kliknout na volbu „Install“. Po instalaci se objeví okno s informací, zda byly nástroje úspěšně nainstalovány.



Obrázek 4.6: JKI VI Package manager po zvolení balíčku

Po nainstalování je možno „VI package manager“ vypnout a spustit LabView. Po otevření nabídky „Funkcí“ je nová možnost nástrojů s názvem „Arduino“, která je zobrazena na obrázku 4.7.



Obrázek 4.7: Nainstalovaný nástroj pro Arduino v programu LabView

#### 4.3.2 Zahájení programování Arduina v Labview

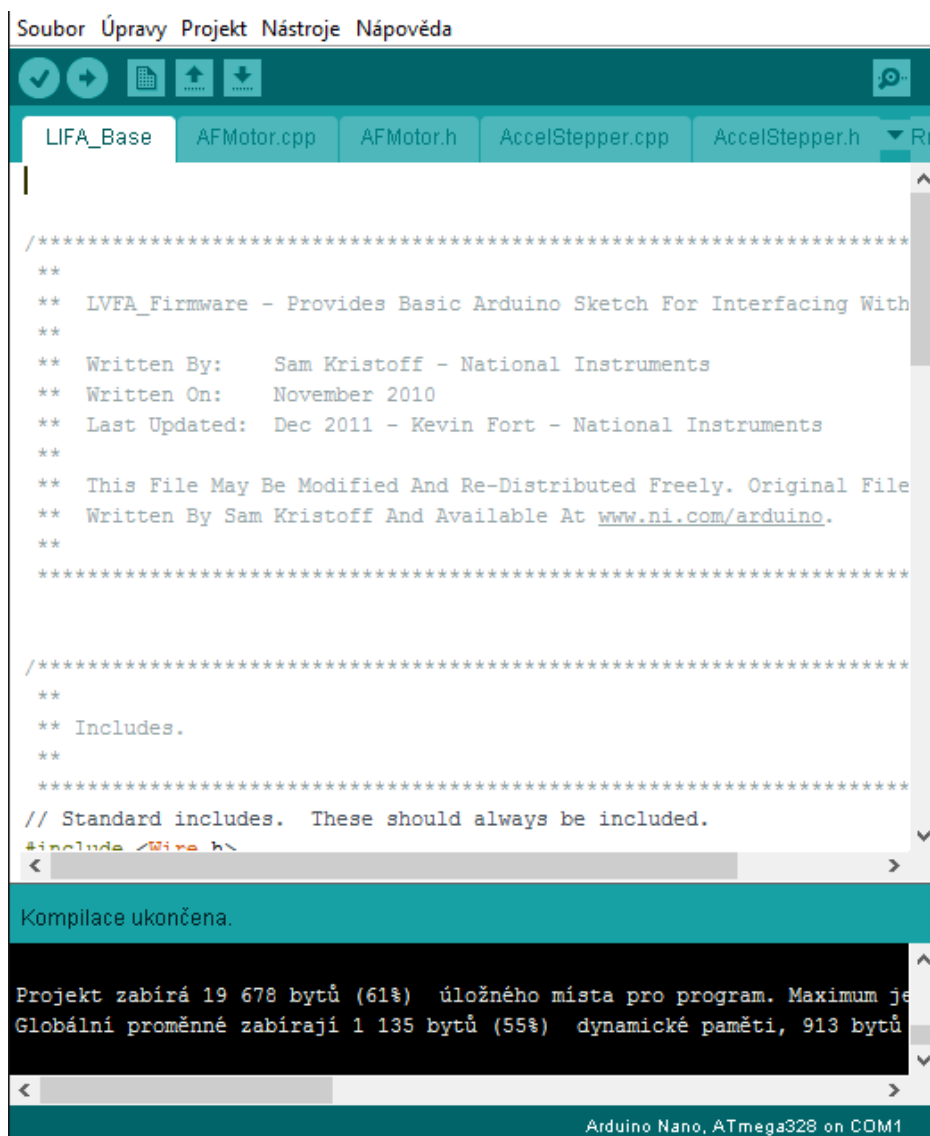
Pro zahájení ovládání desky Arduino přes LabView, je nutné nahrát do Arduina program, který reaguje na příkazy LabView přes USB a správně odesílá požadovaná data. Tento program je stažen společně s balíčkem „LabView Interface for Arduino“, viz kapitola 4.3.1.

Pro nalezení tohoto programu je potřeba znát uložení nainstalovaného LabView. Příklad uložení: „C:\Program Files (x86)\National Instruments\LabVIEW 2013“.

Po otevření složky „LabVIEW 2013“, popřípadě pojmenované jinou verzí LabView, ve které máme nainstalovaný balíček pro „LabView Interface for Arduino“, zde nalezneme cestu: „\vi.lib\LabVIEW Interface for Arduino\Firmware\LIFA\_Base“.

Pokud je známá adresa, je potřeba spustit Arduino IDE, viz kapitola 4.2. V tomto prostředí je nutné nahrát soubor „LIFA\_Base.ino“ ze složky „LIFA\_Base“.

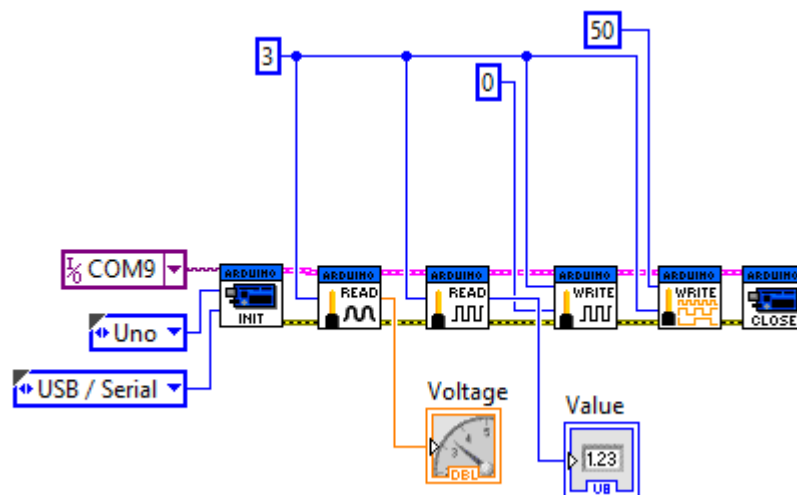
Po otevření tohoto programu je vhodné program zkontrolovat. Správná kontrola a otevření programu je zobrazeno na obrázku 4.8. Dále stačí vybrat používané Arduino a port připojení, jehož vybrání je zobrazeno v pravém spodním rohu na obrázku 4.8. Nakonec stačí program nahrát do Arduina. Po úspěšném provedení lze používat Arduino s LabView.



Obrázek 4.8: Otevření a kompilace LIFA Base.ino

### 4.3.3 Základy programování Arduina v LabView

Základními prvky pro ovládání Arduina je inicializace Arduina, čtení analogového vstupu, čtení digitálního vstupu, zapisování na digitální výstup, zapisování PWM výstupu a uzavření Arduina. Tyto komponenty jsou zobrazeny ve zdrojovém kódu 1.



Zdrojový kód 1: Základní prvky Arduina pro programování v LabView

Inicializace Arduina je potřebná pro navázání komunikace s Arduinem. Je zde potřeba nastavit správný port. Ve zdrojovém kódu 1 je vidět nastavený port „COM9“. Dále je dobré zkontrolovat správný typ připojení a zvolené Arduino. Pro použití desky typu Arduino Nano je potřeba mít zvolený typ desky „Uno“.

Výstup z inicializace je růžová datová linka, která obsahuje informace o zařízení a udává dalším komponentám, odkud mají číst nebo kam zapisovat hodnoty. Druhým výstupem je žlutá datová linka, která obsahuje informace o správnosti průběhu komponenty. Pokud průběh komponenty z nějakého důvodu selže, objeví se zde takzvaný „ERROR“, který informuje, kde a k jakému selhání došlo. Tyto dvě výstupní linky obsahuje každá komponenta využívající Arduino jako dva další vstupy a výstupy.

Čtení analogového vstupu je druhá komponenta zobrazená ve zdrojovém kódu 1. Tato komponenta čte napěťový vstup v rozmezí 0 V až 5 V. Na vstupu této komponenty je číslo analogového pinu na desce Arduino. Výstup je přečtená napěťová hodnota z daného pinu v podobě reálného čísla. Ve zdrojovém kódu 1 je vidět čtení analogového pinu číslo 3 a zapisování do hodnoty „Voltage“.

Čtení digitálního vstupu je podobné jako čtení analogového vstupu s rozdílným výstupem, kde na výstupu se objevuje hodnota 1 nebo 0.

Zapisování na digitální výstup je čtvrtá komponenta ve zdrojovém kódu 1. Tato komponenta má dva vstupy pro určení pinu, na který se má zapisovat a jaká hodnota se má zapsat.

Zapisování na PWM výstup je pátá komponenta ve zdrojovém kódu 1. Jedná se o zapisování hodnoty na digitální pin Arduina. K této možnosti slouží pouze některé piny na desce Arduino. Arduino nano má pro možnost PWM výstupu piny 3, 5, 6, 9, 10 a 11. Jedná se o nastavení hodnoty logické 1 v časovém poměru k logické 0 v jednom cyklu. Komponenta obsahuje vstup pro zvolení digitálního pinu a vstup pro určení poměru, který se zadává v hodnotách 0 až 255. Pokud je nastavená nula, je PWM výstup logická 0, pokud chceme polovinu cyklu mít logickou 1 a polovinu logickou 0 je potřeba nastavit hodnotu 128. Tato možnost nastavování umožňuje například ovládání servopohonů.

Poslední komponentou ve zdrojovém kódu 1 je uzavření Arduina. Tato komponenta je nutná pro správné ukončení funkce desky Ardino a vymazání paměti. Bez správného ukončení je potřeba v desce Arduino před dalším použitím vymazat paměť, o co se stará blok inicializace Arduina, popřípadě restartovat desku mechanicky. Pro mechanické restartování desky slouží fyzické tlačítko na vrchu desky.

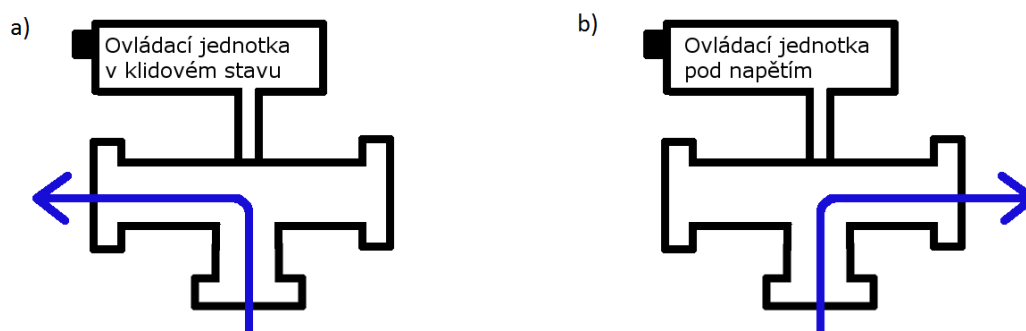


## 5 Připojení komponent k Arduino

Pro propojení komponent s Arduinem bylo nejdříve využité nepájivé pole, na kterém bylo propojení testováno.

### 5.1 Třicestné ventily

Jedná se o zónové ventily s vratnou pružinou, kde po přivedení napájecího napětí 230V se změní průtok kapaliny. Průtok kapaliny ventilem je naznačen na následujícím obrázku, kde v levé části je ventil v klidovém stavu a v pravé části je naznačen průtok, pokud se přivede do ovládací jednotky napájecí napětí 230V.



Obrázek 5.1: Průtoky kapaliny třicestným ventilem – a) klidový stav; b) pod napětím

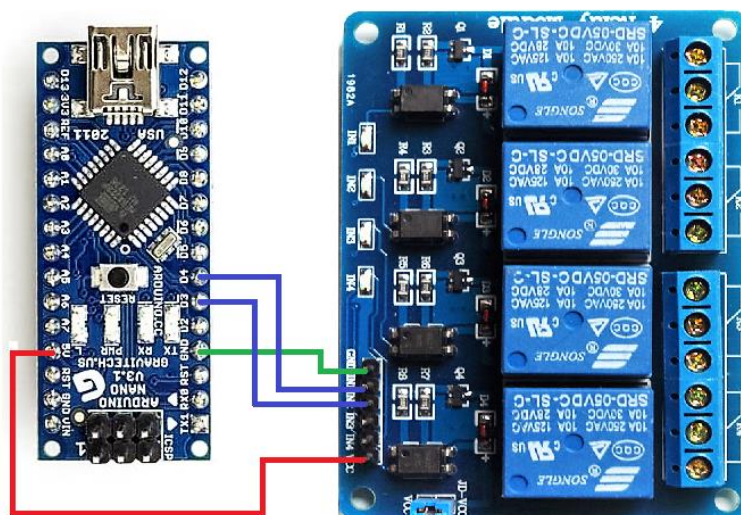
Ovládací jednotka ventilu má dvě funkce. Hlavní funkce je přepínání průtoku pomocí přivedení napájecího napětí 230V. K této funkci zde slouží fázový, nulový a ochranný vodič. Vedlejší funkce je indikace krajní otevřené polohy ventilu. Tato poloha nastane po deseti sekundách při přivedení napájecího napětí 230V. Indikace je zvolena sepnutím spínače v ovládací jednotce, kde vývody ze spínače jsou dostupné jako vodiče vedoucí z ovládací jednotky.

Pro ovládání napájecího napětí byl zvolen modul pro Arduino se spínacími relé. Pro znázornění propojených kontaktů Arduina a spínacího relé modulu slouží tabulka 2 a graficky znázorňuje obrázek 5.2. Pro napájení relé modulu bylo použito výstupní napájení z Arduina.

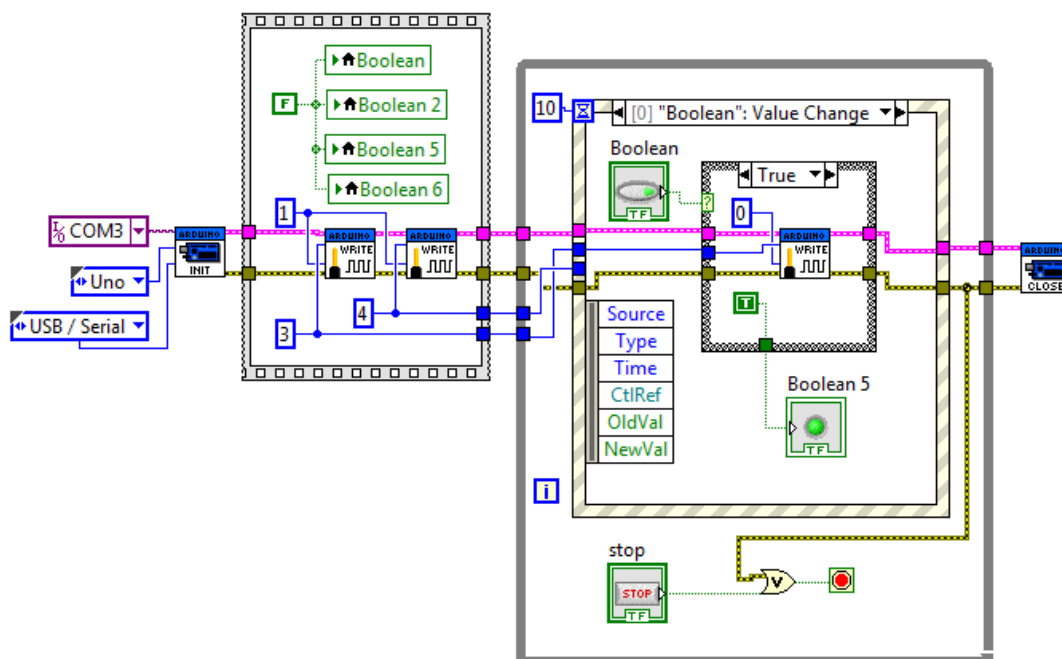
Na kontakty relé se připojilo napájení 230V a napájecí vstup třicestného ventilu. Plné otevření a uzavření ventilu je hlídáno programově, díky dané časové konstantě pro uzavření ventilu 6s a pro otevření ventilu 10s. Základní program pro spínání relé je zobrazen ve zdrojovém kódu 2. Tento program inicializuje Arduino, nastaví základní hodnoty a následně reaguje na akci uživatele, kde po stisknutí tlačítka sepne, nebo rozezne stykač.

### Tabulka 2: Propojení konektorů relé modulu a Arduina nano

<b>Dvojité relé modul</b>	<b>Arduino nano</b>
GND	GND
VCC	5V
IN0	D3
IN1	D4



Obrázek 5.2: Propojení Arduina a dvou relé v relé modulu



### Zdrojový kód 2: Ovládání module relé v programu LabView

## 5.2 Snímače průtoku

Snímače průtoku byly zvoleny od německého výrobce Boi-Tech s označením FCH-m-POM-LC Art.-Nr: 150392. Tento typ průtokoměru dokáže podle údajů od výrobce měřit průtok od 0,05 l/min do 3 l/min. Měření nižšího a přesnějšího průtoku lze dosáhnout použitím přiložené trysky. Protečené množství je udáváno pomocí obdélníkových pulzů, které jsou generovány díky Hallovi senzoru a otevřenému kolektoru na NPN tranzistoru. Počty pulzů a změny s použitím trysky jsou zobrazeny v tabulce 3. Pro napájení průtokoměrů byl zvolen externí napájecí zdroj s napětím 24V a proudem 1A. K vyhodnocení počtu impulzů byla zvolena varianta s čítačem napěťových impulzů. Pro správné použití bylo potřeba přiřadit mezi napájecí vstup a výstupní signál průtokoměru „pull up“ rezistor. Tento rezistor zabraňuje neočekávaným stavům, ke kterým může docházet pomocí otevřenému kolektoru.

Tabulka 3: Parametry průtokoměru [15]

	<b>Průtokoměr bez trysky</b>	<b>Průtokoměr s tryskou</b>
Minimální průtok	0,05 l/min	3,0 l/min
Maximální průtok	0,03 l/min	1,8 l/min
Počet pulzů na litr	2 500	8 500

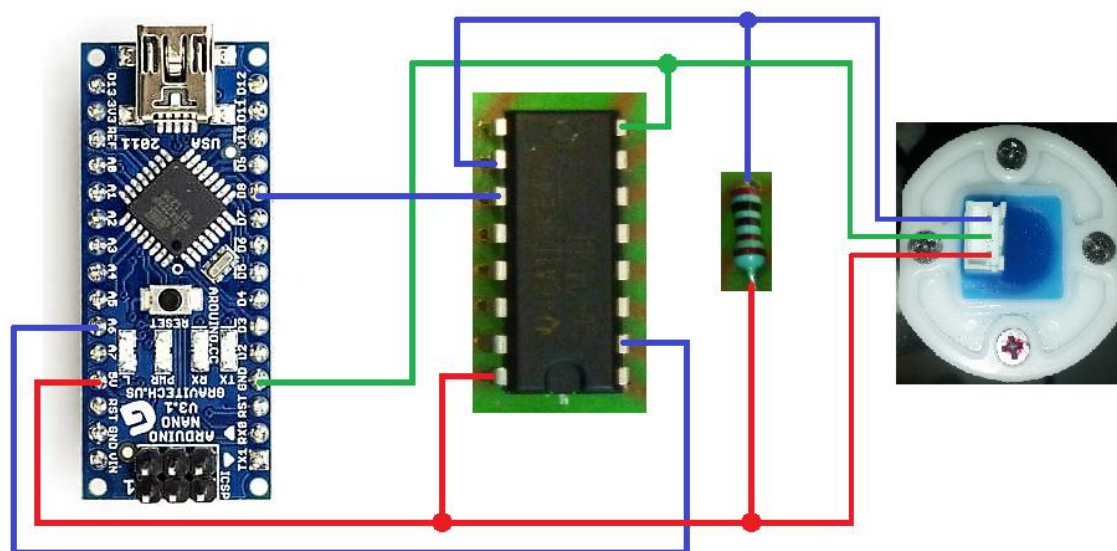
Pro měření počtu pulzů byl použit čítač s označením CD4040BE. Tento čítač čítá napěťové pulzy s binárním výstupem čtrnácti hodnot. Pro napájení čítače bylo použito výstupní napájení z Arduina.

Při zapojení výstupních hodnot z čítače do digitálních vstupů Arduina bylo zjištěno hraniční napětí a tím špatné vyhodnocování počtu impulzů. Z tohoto důvodu byly využity analogové vstupy Arduina, kde se programově vyhodnocuje napětí logické jedničky nad 4 V.

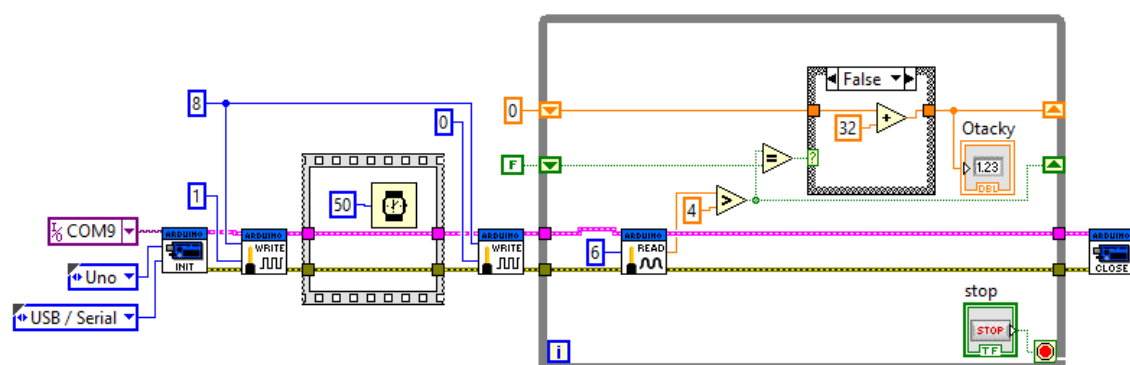
Z důvodu nedostatečného počtu analogových vstupů na desce Arduino nano jsou využity dvě desky Arduino nano, kde každá má k dispozici pět analogových vstupů pro zjišťování hodnot na čítači. Dále je využita možnost vynulování čítače přivedením logické jedničky na nulovací vstup čítače, kde je použit jeden digitální výstup z Arduina. Pro znázornění propojených kontaktů Arduina a čítače s využitím pouze jednoho binárního výstupu slouží tabulka 4 a graficky znázorňuje obrázek 5.3. Program pro jednoduché čtení a zjišťování protečeného množství bez potřebných zabezpečení, jako je například rychlejší změna čítače než samotné čtení, je zobrazen ve zdrojovém kódu 3. Zde po inicializaci Arduina proběhne vynulování čítače a poté program zjišťuje změnu na pinu A6. Pokud dojde ke změně napětí, připočítá 32 otáček, což odpovídá šestému binárnímu výstupu na čítači.

Tabulka 4: Propojení konektorů průtokoměru a Arduina nano

Průtokoměr	Arduino nano	Čítač CD4040BE	Rezistor
+VCC	5V	VDD	Vstup
GND	GND	VSS	
Signal		Input pulses	Výstup
	D8	Reset	
	A6	Q6	



Obrázek 5.3: Propojení Arduina a průtokoměru



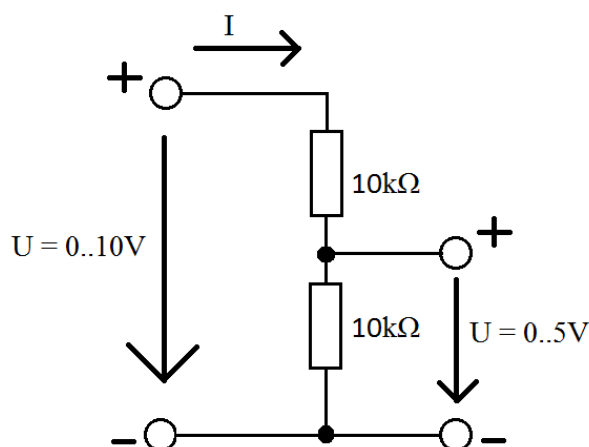
Zdrojový kód 3: Čtení počtu otáček průtokoměru v programu LabView

### 5.3 Snímače tlaku

Pro snímání tlaku se zvolily snímače relativního tlaku s keramickou oddělovací membránou s označením D2410. Tyto tlakoměry se dělí podle dalšího kódu na různé rozsahy. Pro účely filtrační jednotky byl zvolen rozsah do 400 kPa, který nese kódové označení „G140“. Jeho přetížitelnost je stanovena na 1,2 MPa.

Výstupní signál pro určené velikosti tlaku je přímo úměrný rozsahu snímače tlaku a velikosti výstupního napětí ze snímače tlaku, které je stanoveno v rozsahu 0 - 10 V.

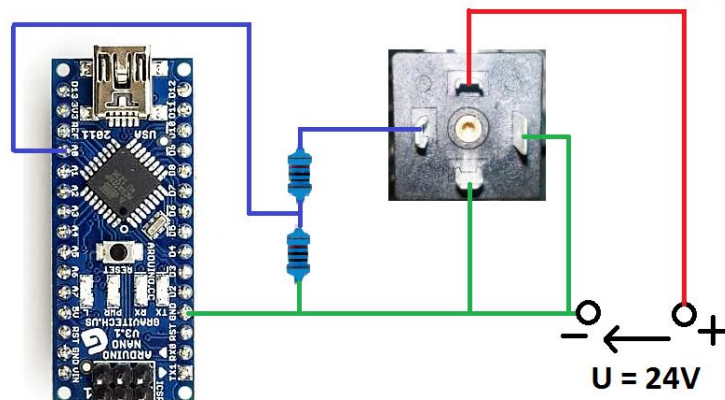
Připojení k analogovému vstupu Arduino nemohlo být přímé, jelikož Arduino podporuje vstupní napětí 0 - 5 V. Pro zmenšení napětí ze snímače tlaku byl použit jednoduchý odporový dělič se dvěma stejnými odpory (10 kΩ). Jednoduché schéma tohoto děliče je znázorněno na obrázku 5.4. Reálné zapojení konektorů uvádí tabulka 5 a fyzicky znázorňuje obrázek 5.5. Jednoduché čtení aktuálního tlaku v LabView zobrazuje zdrojový kód 4.



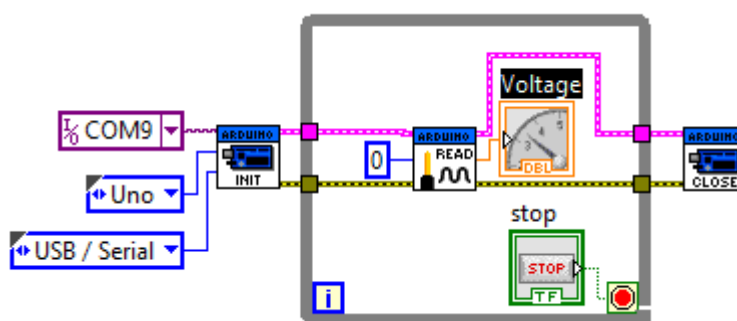
Obrázek 5.4: Schéma jednoduchého odporového děliče

Tabulka 5: Propojení konektorů tlakoměru a Arduino nano

Tlakoměr	Arduino Nano	Odporový dělič	Externí napájení
Napájení +			+ 24V
Kostra a Napájení -	GND	Výstupní / Vstupní -	GND
Signál +		Vstupní +	
	A0	Výstupní +	



Obrázek 5.5: Propojení Arduina a tlakoměru



Zdrojový kód 4: Čtení hodnoty z tlakoměru v programu LabView

## 5.4 Ovládání čerpadla

Čerpadlo bylo ovládáno pomocí ručního ovladače, na kterém se nastavoval chod čerpadla vpřed, nebo zpět a rychlost čerpání. Tento ovladač byl oddělen kvůli umožnění automatického ovládání pomocí Arduina.

Ruční ovládání bylo realizováno pomocí třípolohového tlačítka pro vypnutí čerpadla a zapnutí chodu čerpadla dopředu nebo dozadu. Toto tlačítko bylo nahrazeno relé modulem, kde první relé ovládá zapnutí, nebo vypnutí čerpadla a druhé relé nastavuje chod čerpadla dopředu nebo dozadu.

Ruční ovládání rychlosti čerpadla bylo realizováno pomocí potenciometru, který pomocí zvyšujícího se odporu zpomaloval rychlost čerpadla. Tento odpor se pomocí potenciometru nastavoval v hodnotách 0 – 10 k $\Omega$ . Tento potenciometr byl nahrazen relé modulem se čtyřmi relé a třemi odpory s hodnotami 2 k $\Omega$ , 3 k $\Omega$ , a 5 k $\Omega$ . Toto zapojení umožňuje při vypnutých stykačích zapojení všech tří odporů do série, kde se tyto odpory sčítají a tvoří celkový odpor 10 k $\Omega$ . Postupné spínání stykačů odpory vyřadí ze série, díky čemuž se dá vytvořit částečná stupnice odporů 0 – 10 k $\Omega$ . Pomocí této řady je vytvořeno krokové ovládání rychlosti čerpadla na stupnici 0 – 10, kde maximální výkon je daný stykačem bez připojení a tím vzniká maximální odpor.

Ovládání stykačů je shodné s ovládáním třicestných ventilů viz kapitola 5.1 na stránce 31.

## 5.5 Ventil se servopohonem

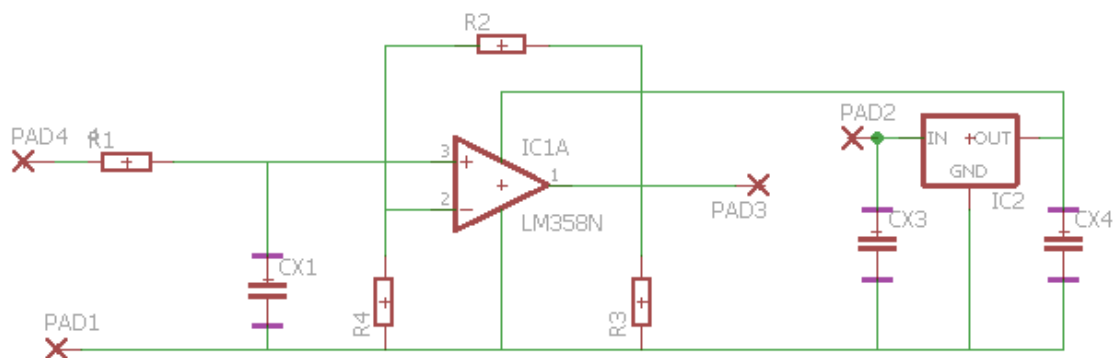
Tento ventil byl přidán pro lepší regulaci tlaku ve filtračním obvodu. Díky němu pro snížení tlaku stačí povolovat ventil, kde plný průchod kapaliny ventilem umožňuje mít v obvodu téměř nulový tlak.

Ventil je ovládaný pomocí otočného servopohonu od výrobce Belimo s označení „LR24S-SR“. Servopohon je řízený pomocí PWM signálu s napětím 0 – 10 V. Jelikož výstup Arduina je maximálně 5 V, bylo zapotřebí zesílit PWM signál. Signál je zesílen pomocí zesilovače LM358N. Napájecí napětí servopohu je 24 V, díky tomu bylo zvolené stejné napájení i pro zapojení zesilovače. Zesilovač vyžaduje napájení 12 V, čehož bylo dosaženo jednoduchým zapojením usměrňovače napětí a dvou kondenzátorů. Hodnoty součástek jsou zobrazeny v tabulce 6 a zapojení je znázorněné v obrázku 5.6.

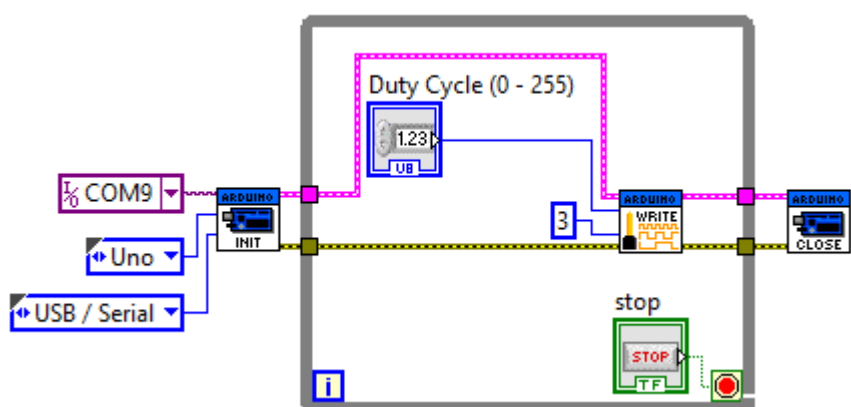
Připojení k Arduinu je možné pouze na PWM digitální výstupy Arduina, kterými jsou digitální piny 3, 5, 6, 9, 10 a 11. Tento pin je připojen na konektor PAD4, viz tabulka 6, a výstup PAD3 je připojen na vstupní signál otočného servopohonu.

Tabulka 6: Součástky pro zesílení PWM signálu

Označení součástky	Hodnota, nebo označení součástky
R1	Odpor 390 k $\Omega$
R2	Odpor 10 k $\Omega$
R3	Odpor 100 k $\Omega$
R4	Odpor 10 k $\Omega$
CX1	Kondenzátor 100 nF
CX3	Kondenzátor 10 $\mu$ F
CX4	Kondenzátor 10 $\mu$ F
IC1A	Zesilovač LM358N
IC2	Usměrňovač napětí L7812CV
PAD1	Zem
PAD2	+ 24V
PAD3	Výstup zesíleného PWM signálu
PAD4	Vstup zesilovaného PWM signálu



Obrázek 5.6: Schéma zapojení součástek v program Eagle



Zdrojový kód 5: Ovládání servopohonu v programu LabView



## 6 Fyzická kompletace filtračního systému

Filtrační systém a jeho komponenty jsou řešeny pomocí speciálního uchycení, které díky mechanickému sevření dvou speciálně upravených konců trubek a vloženého těsnění vytvoří pevný spoj. Toto uchycení je zobrazeno na obrázku 6.1. Díky tomuto speciálnímu uchycení bylo potřeba vyrobit přechody na všechny nově přidané komponenty.



Obrázek 6.1: Uchycení komponent na filtračním zařízení

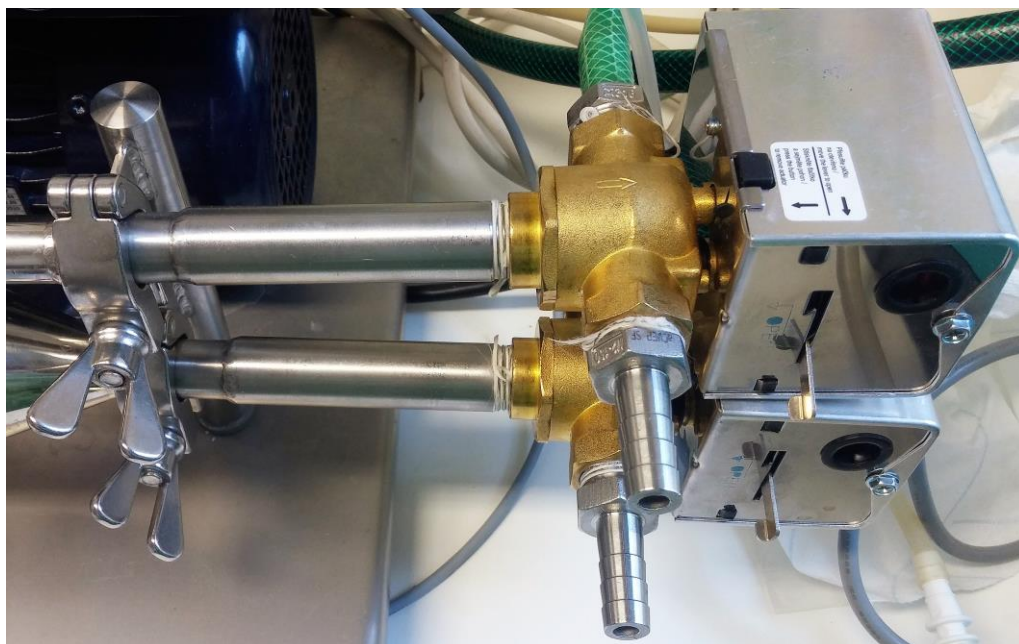
Celkový pohled na filtrační jednotku je zobrazen v příloze č. 11 a v příloze č. 12.

### 6.1 Instalace komponent na filtrační zařízení

Pro vhodné připojení komponent k filtračnímu zařízení bylo potřeba vysoustružit přesné kopie speciálních konců trubek a navařit na potřebné nátrubky.

#### 6.1.1 Instalace třicestných ventilů

K třicestným ventilům bylo zapotřebí vysoustružit na jeden konec nátrubků speciální uchopení k filtrační jednotce. Zbylé potrubní vývody se daly dokoupit a kabely pro ovládání třicestných ventilů byly připojené. Uchycení ventilů je zobrazeno na obrázku 6.2.



Obrázek 6.2: Fyzické uchycení třicestných ventilů

### 6.1.2 Instalace snímačů průtoku

Tyto snímače jsou k filtračnímu zařízení připojené pouze pomocí hadiček. Díky malému průtoku není požadované jejich pevné uchycení. Pro napájení a výstupní signál byly napájeny třívodičové konektory. Tyto průtokoměry a jejich umístění je vidět na obrázku 6.3.



Obrázek 6.3: Umístění průtokoměrů

### 6.1.3 Instalace snímačů tlaku

Tyto snímače jsou zakončené závitem. Pro jednodušší výrobu speciálního uchycení se zvolila varianta navaření nátrubku na vysoustružené speciální uchycení a spojení tohoto celku se snímačem byl dokoupen přechod, který má na každé straně vnitřní závit. Pro napájení a výstupní signál byli připojené čtyřvodičové kabely. Této kompletace je zobrazena na obrázku 6.4.



Obrázek 6.4: Fyzický vzhled snímačů se speciálním uchycením

#### 6.1.4 Instalace ventilu se servopohonem

Pro připojení ventilu se servopohonem bylo potřeba přizpůsobit jednu trubku z původního filtračního systému. Tato trubka byla provrtána a byl zde navařen vyfrézovaný nátrubek. Díky velké odolnosti navaření a pevnému uchycení servopohonu na ventil nebylo potřeba dalšího uchycení. Připojení servopohonu k trubce filtračního systému je zobrazeno na obrázku 6.5.



Obrázek 6.5: Fyzické připojení servopohonu

## 6.2 Návrh plošných spojů – Eagle

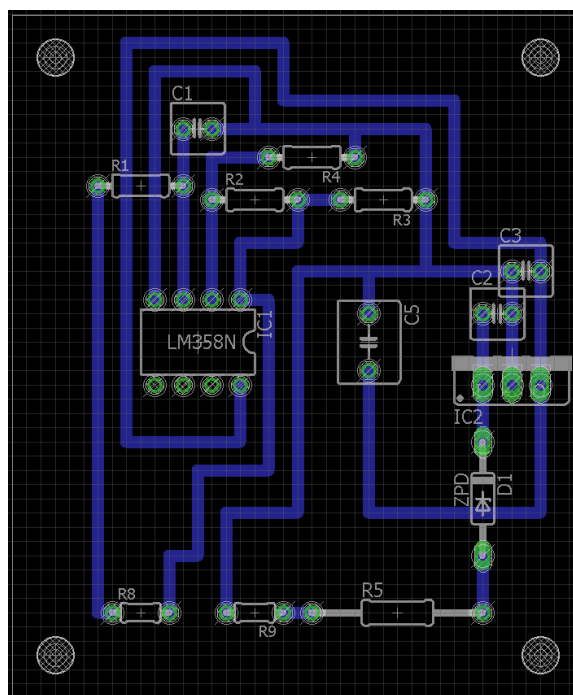
Elektrické obvody pro potřebu propojení Arduina a nových komponent připojených k filtrační jednotce jsou navrženy v programu Eagle.

Byly navrženy dva plošné spoje. První plošný spoj obsahuje elektrický obvod pro zesílení PWM regulace. Tento obvod je popsán v kapitole 5.5 a podoba plošného spoje v programu Eagle je znázorněna na obrázku 6.6. Pro napájecí kabely je zde určena součástka R9, pro vstup a výstup signálu součástka R8 a součástka R5 umožňuje přidání pojistky.

Druhý plošný spoj obsahuje čítač, viz kapitola 5.2 na stránce 33, a odporový dělič, viz kapitola 5.3 na stránce 35. V tomto druhém spoji jsou zabudované dvě lišty pro jednodušší rozvedení napájení. Díky potřebě dvou čítačů a dvou odporových děličů byly tyto plošné spoje vytvořeny dvakrát, díky čemuž je na jednom plošném spoji rozváděné napájení 5 V, na druhém 24 V a uzemnění je společné. Druhý plošný spoj je znázorněn na obrázku 6.7.

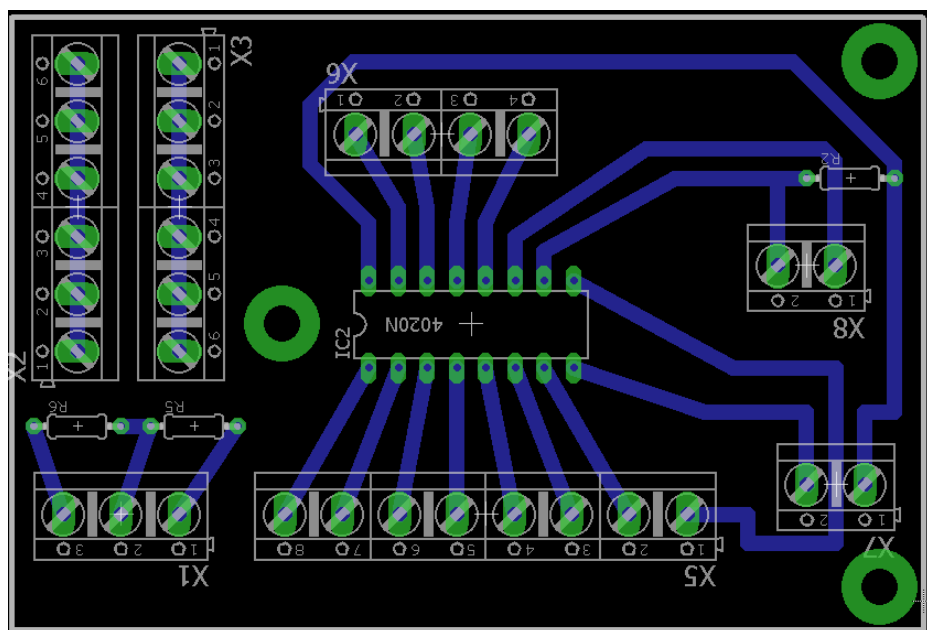
Uchycení kabelů mezi plošnými spoji a Arduinem bylo zvoleno pomocí lišt se šroubky. Díky této volbě je možné snadno měnit, které výstupní hodnoty z čítače bude Arduino využívat. Dále je jednodušší výměna v případě využití jiného plošného spoje.

Pohled na plošné spoje je zobrazen v příloze č. 14.



Obrázek 6.6: Plošný spoj pro PWM regulaci

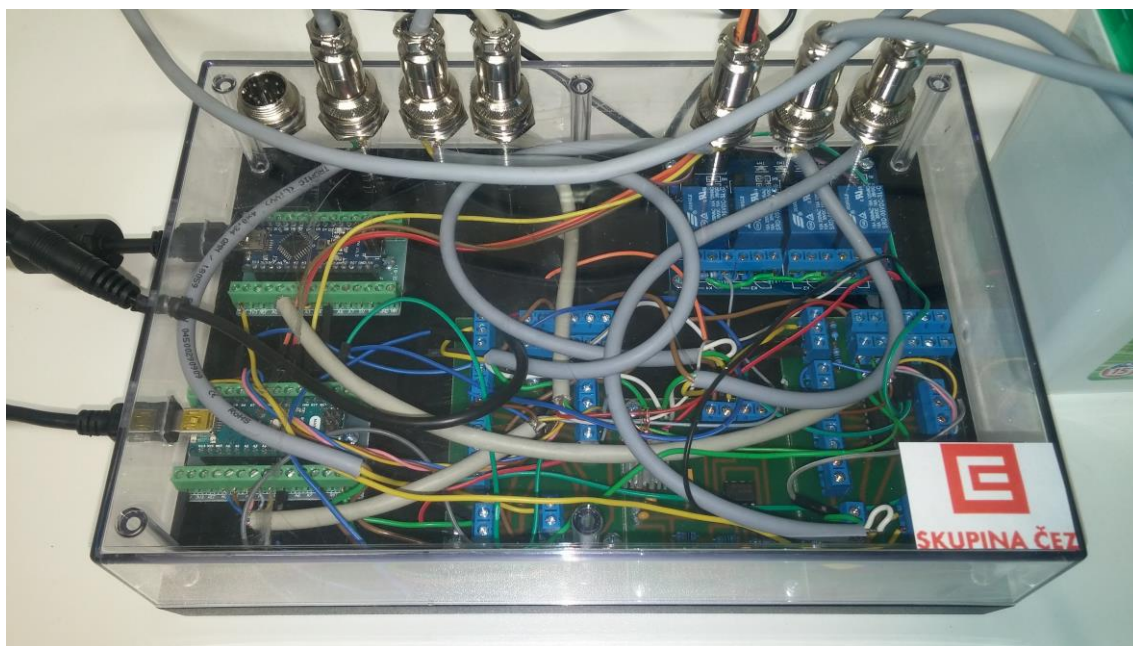




Obrázek 6.7: Plošný spoj pro čítač, odporový dělič a rozvody

### 6.3 Ochranný obal

Ochranný obal je zvolen vodotěsný, pro případ zajišťování vodotěsnosti celého zařízení. Aktuální požadavky byly vedeny na nižší cenu. Z tohoto důvodu jsou použity standardní čtyř a osmipinové konektory pro možnost odpojení elektrických obvodů od filtrační jednotky. Pro připojení Arduina a externího napájení jsou zde otvory zabezpečené vulkanickou páskou, díky čemuž se zvětšuje odolnost vůči vodě a snižuje možnost odpojení kabelů. Tento ochranný obal je zobrazen na obrázku 6.8.



Obrázek 6.8: Ochranný obal

## 7 Vizuální podoba programu

Kompletní program je koncipován pro lehké upravování dle budoucích požadavků uživatele. Základní program v uživatelském rozhraní ukazuje většinu získávaných hodnot, které pro konečného uživatele nejsou potřeba a jsou spíše informativní. Nejdůležitějším údajem, který je zaznamenáván i do výstupního souboru, je výsledná permeabilita, čas jejího měření a celkové protečené množství permeátu. Další z prvků uživatelského rozhraní je ovládání komponent filtrační jednotky, které se bez přímých údajů z provozu nedá přesně stanovit. Z tohoto důvodu je uživatelské rozhraní v základním programu uskutečněno ovládání všech dostupných komponent.

### 7.1 Ovládací panel programu



Program je aktuálně nastaven pro zahájení měření po spuštění tlačítka „Spustit měření“. Potřebnými údaji na začátku měření je pouze „Čas mezi vážením“ a „Počet membrán“. Zbylé nastavení, popřípadě zapnutí řízení podle tlaku či průtoku, lze libovolně měnit při běhu měření. Ovládací panel je zobrazen na obrázku 7.1.



Obrázek 7.1: Ovládací panel programu filtrační jednotky

### 7.2 Zobrazení měřených hodnot

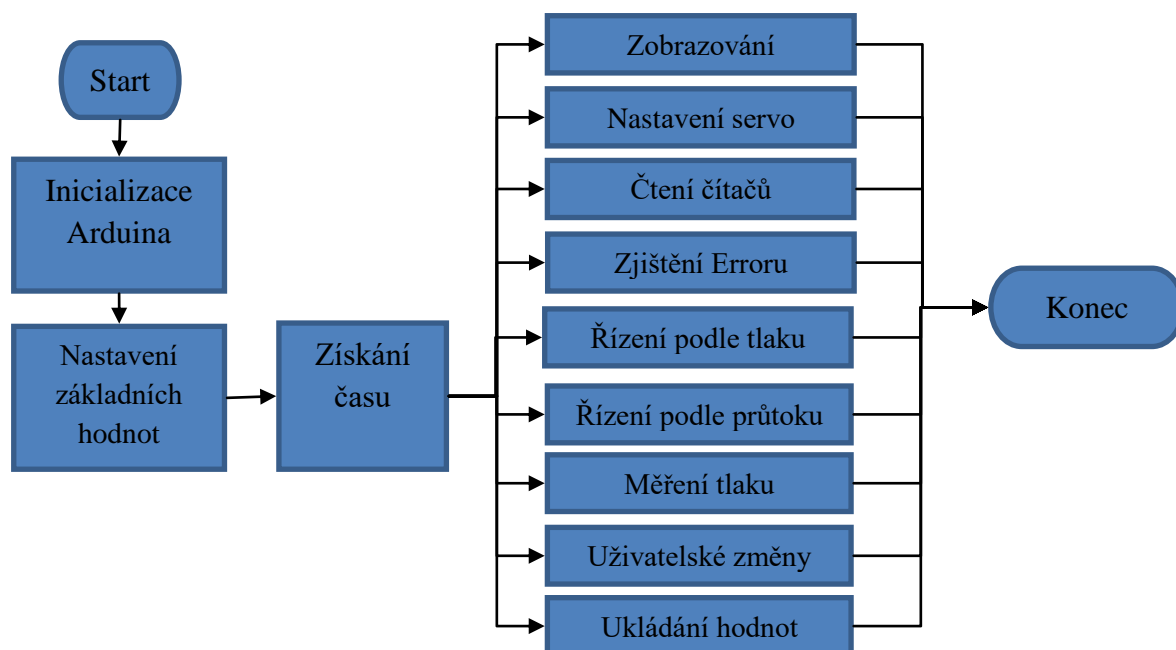
Naměřené, nebo vypočítané hodnoty jsou zobrazeny v druhé části uživatelského rozhraní, které je zobrazeno na obrázku 7.2. Celkový průběh vypočítané permeability je zobrazen ve dvou grafech, který je viditelný v příloze č. 2.

Datum spuštění 02.05.2016		Čas spuštění 14:13		Vypnout měření 		Nulování tlaku 	
Tlak Aktuální [Bar] 0,63504	Tlak Průměrný[Bar] 0,280628	Tlak 2 Aktuální [Bar] 0,61936	Tlak 2 Průměrný [Bar] 0,28614				
Otacky 1 768	Počet litru 1 0,1027	Otacky 2 224	Počet litru 2 0,02995				
Aktuální permeabilita 177,578	Aktuální Průtok 7,67164	Aktuální permeabilita 2 57,1035	Aktuální Průtok 2 2,55721				

Obrázek 7.2: Zobrazovací panel hodnot programu filtrační jednotky

## 8 Program pro ovládání filtrační jednotky

Program je koncipován pro možnost snadného a rychlého upravování v případě nutnosti změny. Hlavní průběh programu je rozdělen do pěti částí, které po správné inicializaci Arduina vždy proběhnou. Hlavní měřicí část je dále rozdělena do několika nezávislých vláken. Tyto části a vlákna jsou zobrazeny na obrázku 8.1.



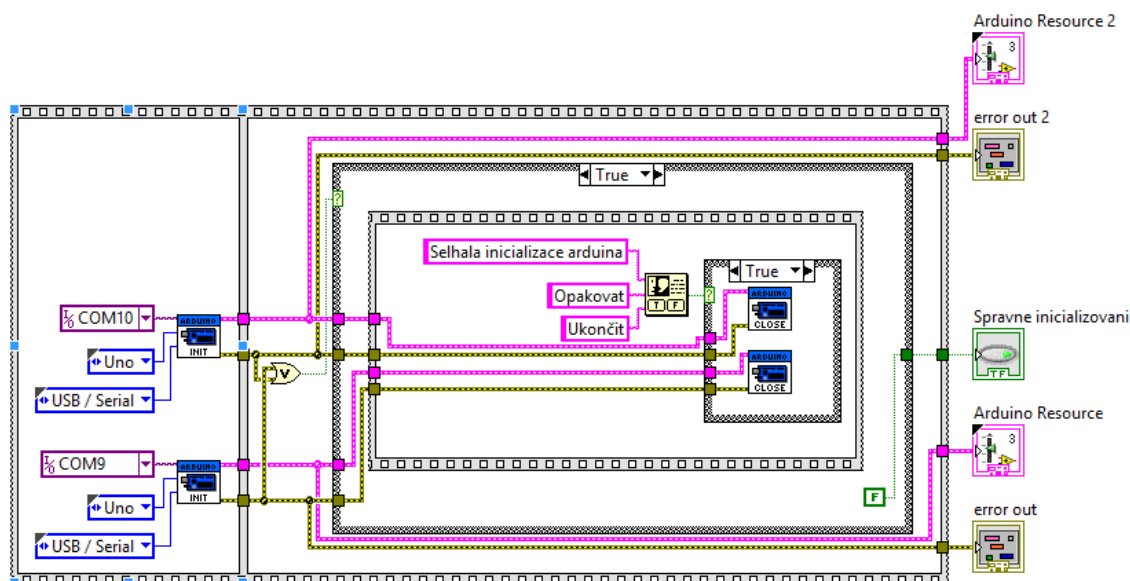
Obrázek 8.1: Graf základních bloků programu filtrační jednotky

### 8.1 První část programu: inicializace Arduina

První část programu nastavuje čas a datum spuštění aplikace, přiřazuje hodnoty pinů, ke kterým jsou připojené fyzické komponenty, a kontroluje správné připojení k Arduinu. Při neúspěšném připojení zobrazí hlášku a umožní vypnutí aplikace, nebo opětovný pokus o spojení. Při opětovném pokusu o spojení se program pokusí správně ukončit Arduino, vymazat jeho paměť a opakuje spojení s Arduinem. Spojení s Arduinem zobrazuje zdrojový kód 6 a celou inicializační část programu ukazuje příloha č. 2.

Inicializace Arduina probíhá na porty COM9 a COM10, kde jejich nastavení je uvedené v kapitole 4.1 na stránce 20.





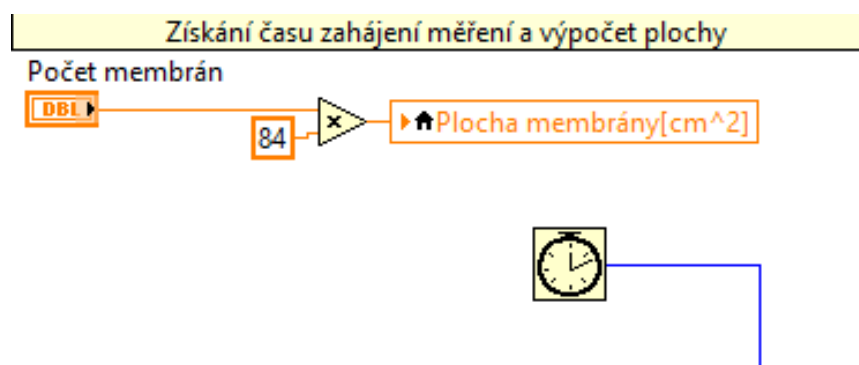
Zdrojový kód 6: Inicializace Arduino

## 8.2 Druhá část programu: inicializace hodnot v Arduino

Ve druhé části programu se nastavují základní hodnoty v Arduino tak, aby například nedošlo k samovolnému spuštění čerpadla. Až v této fázi bude dle návodu možno připojit kompletní napájení filtrační jednotky. Tato část programu je zobrazena v příloze č. 4.

## 8.3 Třetí část programu: získání času zahájení měření

Jedná se o nejkratší část celého programu, kde se vypočítá plocha membrány a zjistí se přesný čas zahájení měření pro následné výpočty. Tato část programu je zobrazena ve zdrojovém kódu 9.



Zdrojový kód 7: Třetí část programu: získání času zahájení

## 8.4 Čtvrtá část programu: průběh měření

Celkový průběh měření je koncipován do několika vláken. Každé vlákno vykonává určitou funkci a všechna vlákna běží paralelně bez závislosti na jiném. Díky této vlastnosti je možné upravovat každou vlastnost měření samostatně.

### 8.4.1 Zobrazování grafů, nastavování servopohonu a čtení počtu otáček

Tato vlákna jsou zobrazena v příloze č. 5.

Zobrazování umožňuje nezávislou vizualizaci dat do grafu. Oddělené zobrazování je vytvořeno z důvodu možnosti předělávání grafů, například jejich zvětšení a umožnění volby zobrazovaných hodnot na jednom větším grafu pro zřetelnější výsledek měření. Tato možnost je předpokládána z předešlého vývoje aplikací.

Nastavování servopohonu probíhá nezávisle od vypočítávání potřebné polohy. Jeho polohu může měnit uživatel ručně, nebo zvolit automatické řízení podle tlaku nebo průtoku.

Čtení počtu otáček je tvořené jednoduchým zjišťováním změny bitu na čítači a upravená výsledná hodnota na celkový počet litrů.

### 8.4.2 Ukončení měření při Erroru a řízení podle tlaku nebo průtoku

Vlákna jsou zobrazena v příloze č. 6.

Ukončení při erroru je vlákno pro registrování chybových hlášek Arduina a předcházení tak zacyklení programu. Při Chybovém hlášení se nastaví ukončovací proměnná, díky které se postupně ukončí všechny smyčky.

Programování režimu s konstantním tlakem je ulehčeno pomocí fyzickému přidání servopohonu a vývodní trubky před tlakoměry. Díky této volbě stačí uvolňovat průchod vody ventilem, pokud je větší tlak v potrubí, než je požadován. Tlak v potrubí se určuje průměrem z obou tlakoměrů. Výsledná hodnota pro nastavení servopohonu se reguluje pomocí PID regulace.

Režim s konstantním průtokem je na stejném principu jako u řízení podle tlaku s rozdílnou rozhodovací podmínkou, která je průměrem z hodnot obou průtokoměrů.

### 8.4.3 Měření tlaku

Výsledkem měření tlaku jsou dvě hodnoty. První udává aktuální hodnotu tlaku v potrubí. Druhá hodnota postupně vypočítává průměrný tlak v čase, kdy se čeká na výpočet permeability. Po výpočtu permeability se vyšle signál pro nulování průměrného tlaku. V tento moment se vynulují hodnoty pro výpočet průměrného tlaku, přenastaví se signál pro vymazání průměrného tlaku a počítá se nová sekvence průměrného tlaku.

Program pro toto měření je zobrazen v příloze č. 7.

#### **8.4.4 Změna hodnot uživatelem**

Volby uživatele na ovládacím panelu jsou hlídány pomocí vlákna, které je zobrazeno v příloze č. 8. Tato část programu zajišťuje obsluhu všech dostupných akčních prvků na ovládacím panelu pomocí struktury událostí, do které se vloží použité akční prvky, a při jejich změně se provede daná vnitřní struktura.

Například při sepnutí tlačítka „vrchní ventil“ se nastaví příslušný pin na Arduino pro sepnutí stykače, ke kterému je připojený třicestný ventil a rozsvítí kontrolku.

#### **8.4.5 Uložení hodnot do pole a výpočet permeability**

Jedná se o nejkomplikovanější část celého programu a je zobrazen v příloze č. 9. Výpočet permeability je protečený objem za určitý čas, který je vynásobený tlakem. Tento výpočet je zobrazen v příloze č. 10.

Toto vlákno se dále stará o hlídání správného času měření, vytváření pole pro výpočet permeability a pro zobrazování grafu, ukládání naměřených a vypočtených hodnot do souboru a předávání pokynu k vymazání průměrného tlaku.

### **8.5 Pátá část programu: ukončení**

Pokud program projde přes první část a nebude nijak externě vypnut, postupně ukončí všechna vlákna ve čtvrté části a v páté části správně uzavře připojení k Arduino.

## 9 Tvorba manuálu

Uživatelský manuál pro filtrační jednotku je přiložen v příloze č. 14. Obsahuje základní pokyny pro instalaci, zprovoznění a ovládání programu pro filtrační jednotku.

V manuálu je podrobně rozebráno:

- a) Základní zásady pro připojení napájení filtrační jednotky.

Kvůli možnosti odpojování a připojování k desce Arduino není možné mít základní nastavení výstupních pinů v potřebné poloze. Díky tomu je možné předčasné sepnutí stykačů a tím zapnutí čerpadla. Dále pak je možné přenastavení servopohonu a tím otevření ventilu pro regulaci.

- b) Propojení kabelů.

Propojovací kabely pro měřicí a ovládací komponenty na filtrační jednotce jsou většinou čtyřvodičové. Kvůli možnosti záměny při zapojování jsou zde popsány konektory a k nim příslušné komponenty filtrační jednotky.

- c) Návod pro připojení Arduino k počítači.

Při prvním spouštění je nutné nastavit příslušné COM porty ke každému Arduino. V této části je přesný popis pro správné nastavení.

- d) Základní spuštění programu.

Obsahuje informace, které parametry jsou nutné nastavit před samotným spuštěním měření a uvádí jejich základní hodnoty.

- e) Popis ovládacích prvků v uživatelském rozhraní.

Zahrnuje všechny použitelné ovládací prvky v uživatelském rozhraní programu pro filtrační jednotku.

- f) Popis základní možné chyby.

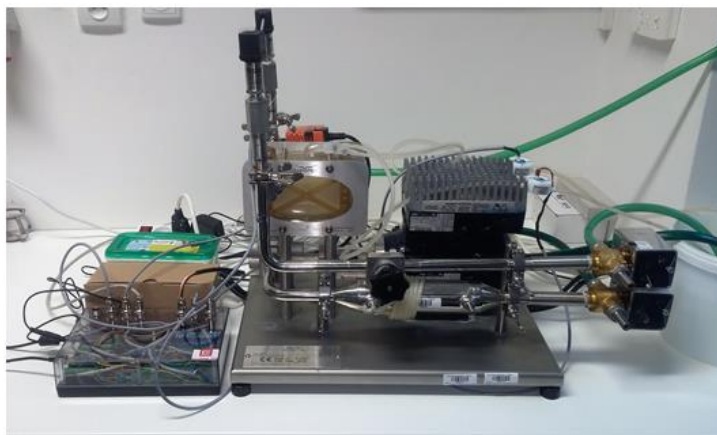
Jedná se o selhání inicializace Arduina, které může být způsobeno různými příčinami. O třech základních příčinách manuál informuje. První příčinou je možnost nepřipojeného Arduina k počítači. Druhá možnost odkazuje na správné nastavení COM portů v počítači. Třetí možnost je standardní opakování, které je vhodné kvůli možnosti špatného ukončení Arduina při předchozím používání.

- g) Popis výstupního souboru.

Popisuje každý sloupec výstupní tabulky z měření.

Titulní list manuálu je na obrázku 9.1.

# Uživatelský manuál pro ovládání filtrační jednotky



## Obsah:

1	Hlavní zásady .....	2
2	Zapojení kabelů filtrační jednotky .....	2
3	Připojení Arduino k počítači .....	3
3.1	Ověření instalace ovladačů .....	3
3.2	Nastavení portů Arduino .....	4
4	Spuštění programu pro filtrační jednotku .....	6
5	Popis tlačítek ovládacího panelu .....	7
6	Chyba: selhání inicializace Arduino .....	8
7	Výstupní soubor .....	9

Obrázek 9.1: Titulní stránka manuálu

## 10 Závěr

Výsledkem této práce je funkční automatická filtrační jednotka, která splňuje všechny stanovené požadavky. Byla vytvořena modifikace pro připojení nových měřících a ovládacích komponent, které navrhl Lukáš Pelc ve svém bakalářském projektu. Dále byla navržena a realizována veškerá elektroinstalace. Ve vývojovém prostředí LabView byl vytvořen program pro Arduino, který ovládá celý systém filtrační jednotky.

Ochranná krabice pro plošné spoje splnila účel jednodušší manipulace a možnosti pověšení na zeď pro případ úniku kapaliny. Do této krabice je navíc připevněn nepoužitý konektor, který umožní v budoucnu provádět další rozšíření, například testování membrán podtlakem. Mechanická konstrukce filtrační jednotky byla při testování dostatečně odolná a při plném výkonu neunikala žádná kapalina. K volbě vstupní a výstupní nádrže s kapalinou jsou určeny dva třicestné ventily, díky kterým lze proplachovat membrány čistou kapalinou. Pro regulaci celého filtračního obvodu byl použit servopohon značky Belimo s kulovým ventilem, který díky PID regulaci dostatečně rychle reguluje celou filtrační soustavu. Program pro filtrační jednotku obsahuje všechny základní funkce potřebné pro ovládání filtračního oběhu a zobrazuje všechny požadované údaje o měření.

Při závěrečném měření s reálnými podmínkami se program choval správně, ovšem při zanešení membrán a s tím spojený pokles výstupního permeátu byla zjištěna nedostatečná přesnost navržených a instalovaných průtokoměrů při menším průtoku permeátu. Z tohoto důvodu budou objednané a nainstalované vhodnější průtokoměry dle přesnějších požadavků.

Zařízení je tedy kompletní funkční celek pro membrány s vyšší propustností a bude dále zpřesňováno pro možnost využití širšího spektra membrán.

Zařízení přinese výhody pro studenty i pracovníky TUL, kteří se zabývají testováním membrán v biotechnické laboratoři. Tuto filtrační jednotku stačí správně nastavit, spustit a výsledné měření se zaznamenává v uloženém dokumentu.

## Seznam použité literatury

- [1] Automatizace měření aneb Když to změří počítač. *Http://docplayer.cz/* [online]. [cit. 2016-03-05]. Dostupné z: <http://docplayer.cz/3419386-Automatizace-mereni-aneb-kdyz-to-zmeri-pocitac.html>
- [2] PRECLÍK, Jan. *POČÍTAČE CESTA OD STAROVĚKU DO KONCE 19. STOLETÍ* [online]. Praha, 2001 [cit. 2016-03-05]. Dostupné z: [http://dml.cz/bitstream/handle/10338.dmlcz/402126/DejinyMat\\_16-2001-1\\_7.pdf](http://dml.cz/bitstream/handle/10338.dmlcz/402126/DejinyMat_16-2001-1_7.pdf)
- [3] Alfa Laval labUnit M10. *Alfalaval.com* [online]. [cit. 2016-04-25]. Dostupné z: <http://www.alfalaval.com/products/separation/membranes/Standalone-pilot-plants/labunit-m10/>
- [4] PELC, Lukáš. *Návrh automatického řízení cirkulačního obvodu filtrace kalových vod*. Liberec, 2015. Bakalářský projekt. Technická Univerzita v Liberci. Vedoucí práce Ing. Lenka Kretschmerová, Ph.D., ING.PAED.IGIP.
- [5] Ventil zónový třicestný s vratnou pružinou VZP 315-230-1P. *Http://www.eshopregulus.cz/* [online;]. [cit. 2016-02-19]. Dostupné z: <http://www.eshopregulus.cz/demved/eshop/15-1-Ventily-zonove-a-smesovaci/182-3-Zonove-ventily-tricestne/5/1854-Ventil-zonovy-tricestny-s-vratnou-pruzinou-VZP-315-230-1P>
- [6] Belimo kulový kohout R2015-P63-S1. *Http://www.elventil.cz/* [online;]. [cit. 2016-02-19]. Dostupné z: [http://www.elventil.cz/200585,cz\\_belimo-kulovy-kohout-r2015-p63-s1-\(r-209\).html](http://www.elventil.cz/200585,cz_belimo-kulovy-kohout-r2015-p63-s1-(r-209).html)
- [7] Servopohon BELIMO-LR24A-SR. *Http://www.elventil.cz/* [online;]. [cit. 2016-02-19]. Dostupné z: [http://www.elventil.cz/23172,cz\\_servopohon-belimo-lr24a-sr.html](http://www.elventil.cz/23172,cz_servopohon-belimo-lr24a-sr.html)
- [8] Průtokoměr B.I.O-TECH e.K. FCH-m-POM-LC. *Http://www.conrad.cz/* [online;]. [cit. 2016-02-19]. Dostupné z: <http://www.conrad.cz/prutokomer-b-i-o-tech-e-k-fch-m-pom-lc-0-03-3-0-l-min.k150392>
- [9] Snímače relativního tlaku D2410. *Http://www.jsp.cz/* [online;]. [cit. 2016-02-19]. Dostupné z: [http://www.jsp.cz/cz/sortiment/seznam\\_dle\\_kategorie/snimace\\_tlaku\\_a\\_tlakove\\_diference/snimace\\_tlaku\\_a\\_tlakove\\_diference\\_s\\_analogovym\\_vystupem/1evne-snimace-relativniho-tlaku-d2410.html](http://www.jsp.cz/cz/sortiment/seznam_dle_kategorie/snimace_tlaku_a_tlakove_diference/snimace_tlaku_a_tlakove_diference_s_analogovym_vystupem/1evne-snimace-relativniho-tlaku-d2410.html)
- [10] *Arduino.cz* [online]. [cit. 2016-02-19]. Dostupné z: <http://arduino.cz/>
- [11] *Arduino.cc* [online]. [cit. 2016-02-19]. Dostupné z: <http://arduino.cc/>
- [12] Arduino Nano. *Https://www.arduino.cc* [online]. [cit. 2016-03-05]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardNano>
- [13] *The Arduino Playground* [online]. [cit. 2016-03-06]. Dostupné z: <http://playground.arduino.cc/>

- [14] *Codeproject.com* [online]. [cit. 2016-03-06]. Dostupné z: <http://www.codeproject.com/Articles/840511/Introduction-to-Netduino-Plus-with-examples>
- [15] Průtokoměr FCH-m-POM-LC. *Conrad.de* [online]. [cit. 2016-04-29]. Dostupné z: <https://www.conrad.de/de/flow-meter-fch-m-pom-lc-003-30-lmin-bio-tech-ek-fch-m-pom-lc-ad-6-mm-mit-duese-16-mm-003-18-lmin-ohne-duese-d-3-mm-005-30-lmin-150392.html>
- [16] Datasheet CD4040BE. *Alldatasheet.com* [online]. [cit. 2016-04-29]. Dostupné z: <http://www.alldatasheet.com/datasheet-pdf/pdf/26879/TI/CD4040BE.html>



## Seznam příloh

<b>Příloha č. 1: Přiložené CD – externí příloha .....</b>	<b>56</b>
<b>Příloha č. 2: Uživatelské rozhraní programu .....</b>	<b>57</b>
<b>Příloha č. 3: První část programu: inicializace Arduina .....</b>	<b>58</b>
<b>Příloha č. 4: Druhá část programu: inicializace hodnot v Arduinu.....</b>	<b>59</b>
<b>Příloha č. 5: Vlákna pro zobrazování, nastavování servopohonu a čtení otáček....</b>	<b>60</b>
<b>Příloha č. 6: Vlákna ukončení při Erroru a řízení podle tlaku nebo průtoku .....</b>	<b>61</b>
<b>Příloha č. 7: Vlákno měření tlaku .....</b>	<b>62</b>
<b>Příloha č. 8: Vlákno pro změnu hodnot uživatelem .....</b>	<b>63</b>
<b>Příloha č. 9: Vlákno pro výpočet permeability a uložení hodnot .....</b>	<b>64</b>
<b>Příloha č. 10: Programový výpočet permeability .....</b>	<b>65</b>
<b>Příloha č. 11: Fotografie celé filtrační jednotky.....</b>	<b>66</b>
<b>Příloha č. 12: Fotografie filtrační jednotky z bočního pohledu.....</b>	<b>67</b>
<b>Příloha č. 13: Fotografie ovládací elektroniky .....</b>	<b>68</b>
<b>Příloha č. 14: Uživatelský manuál – externí příloha.....</b>	<b>69</b>

## **Příloha č. 1: Přiložené CD – externí příloha**

Součástí bakalářské práce je přiložené CD obsahující tyto adresáře:

- **Text**

tato práce v digitální podobě a uživatelský manuál.

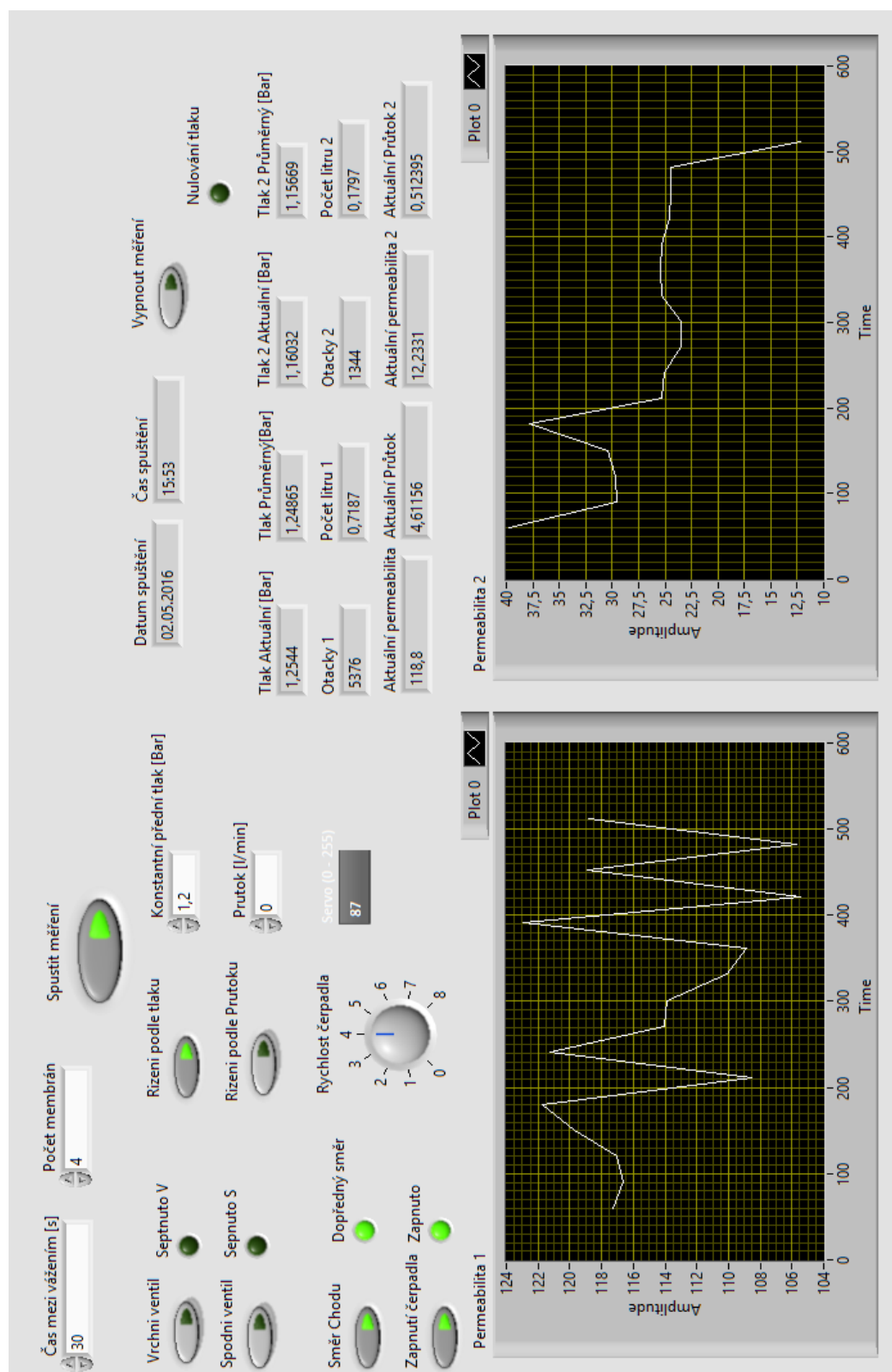
- **Dokumentace fyzických komponent**

dokumentace k použitým elektrotechnickým zařízením.

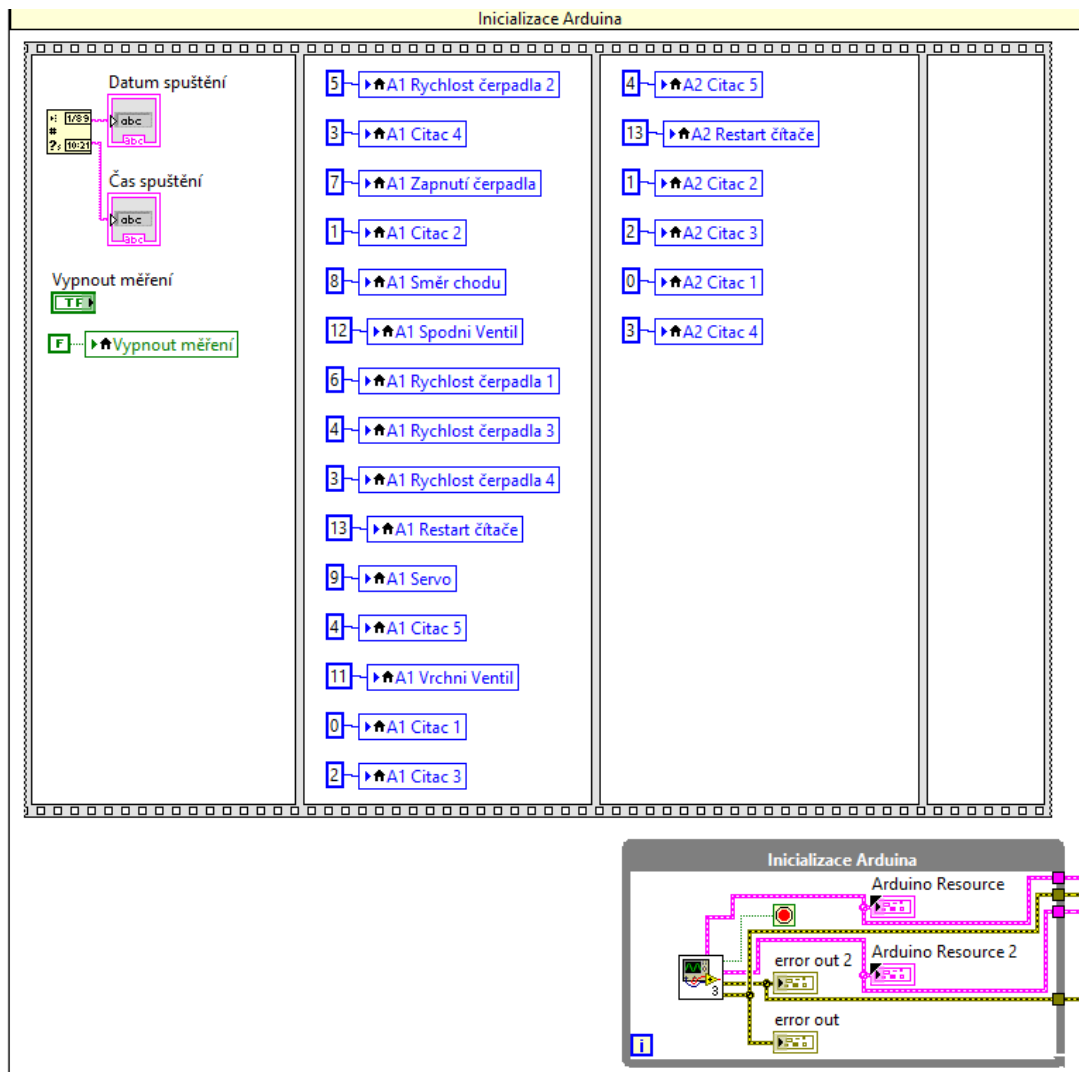
- **Software pro filtrační jednotku**

kompletní program spustitelný v LabView.

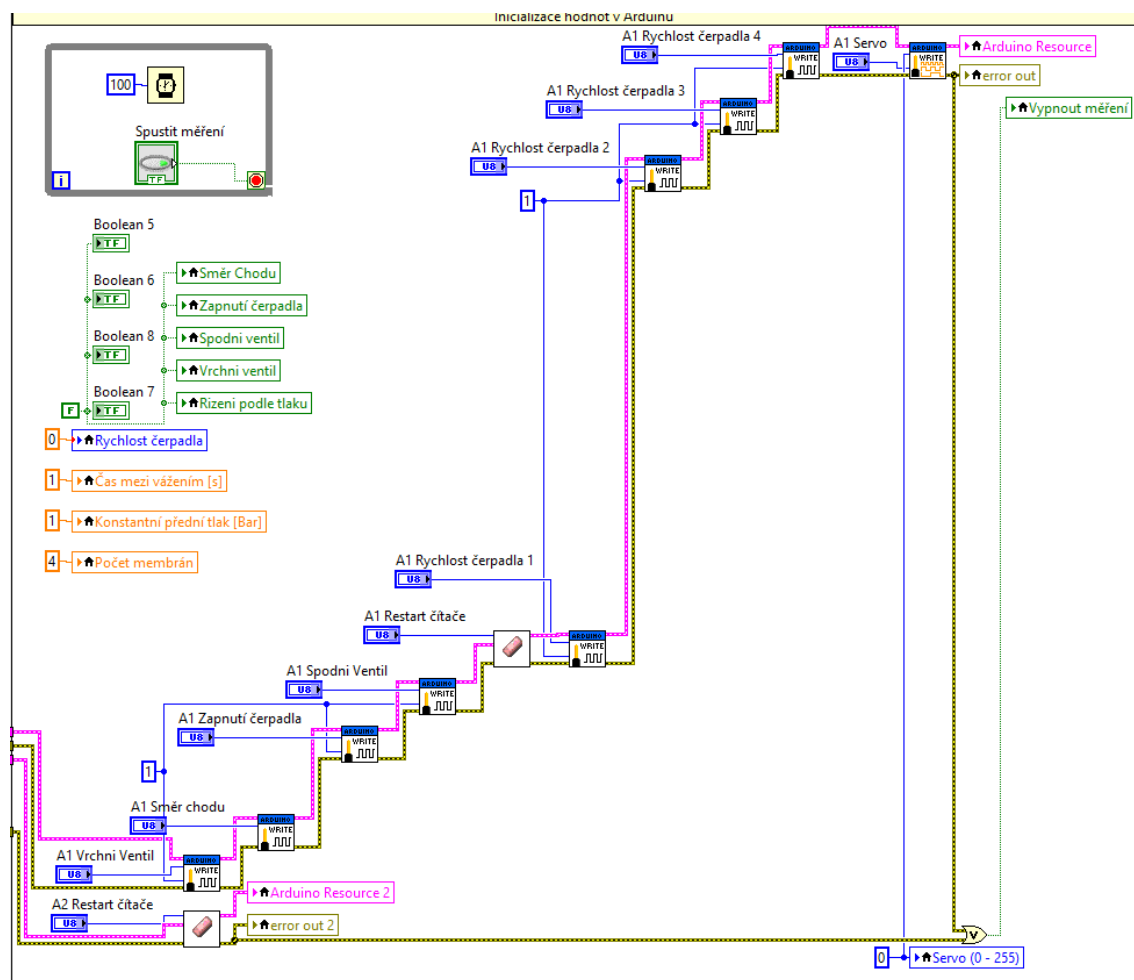
## Příloha č. 2: Uživatelské rozhraní programu



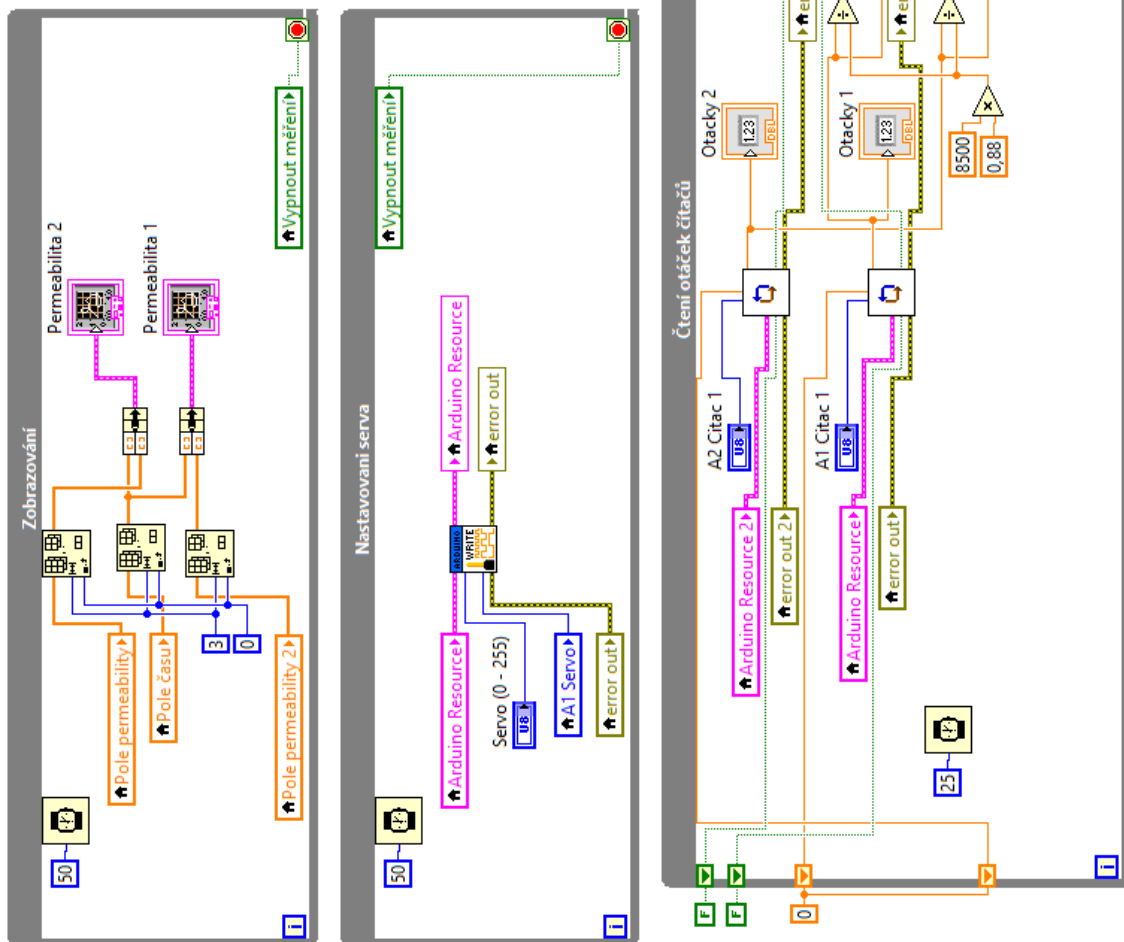
### Příloha č. 3: První část programu: inicializace Arduina



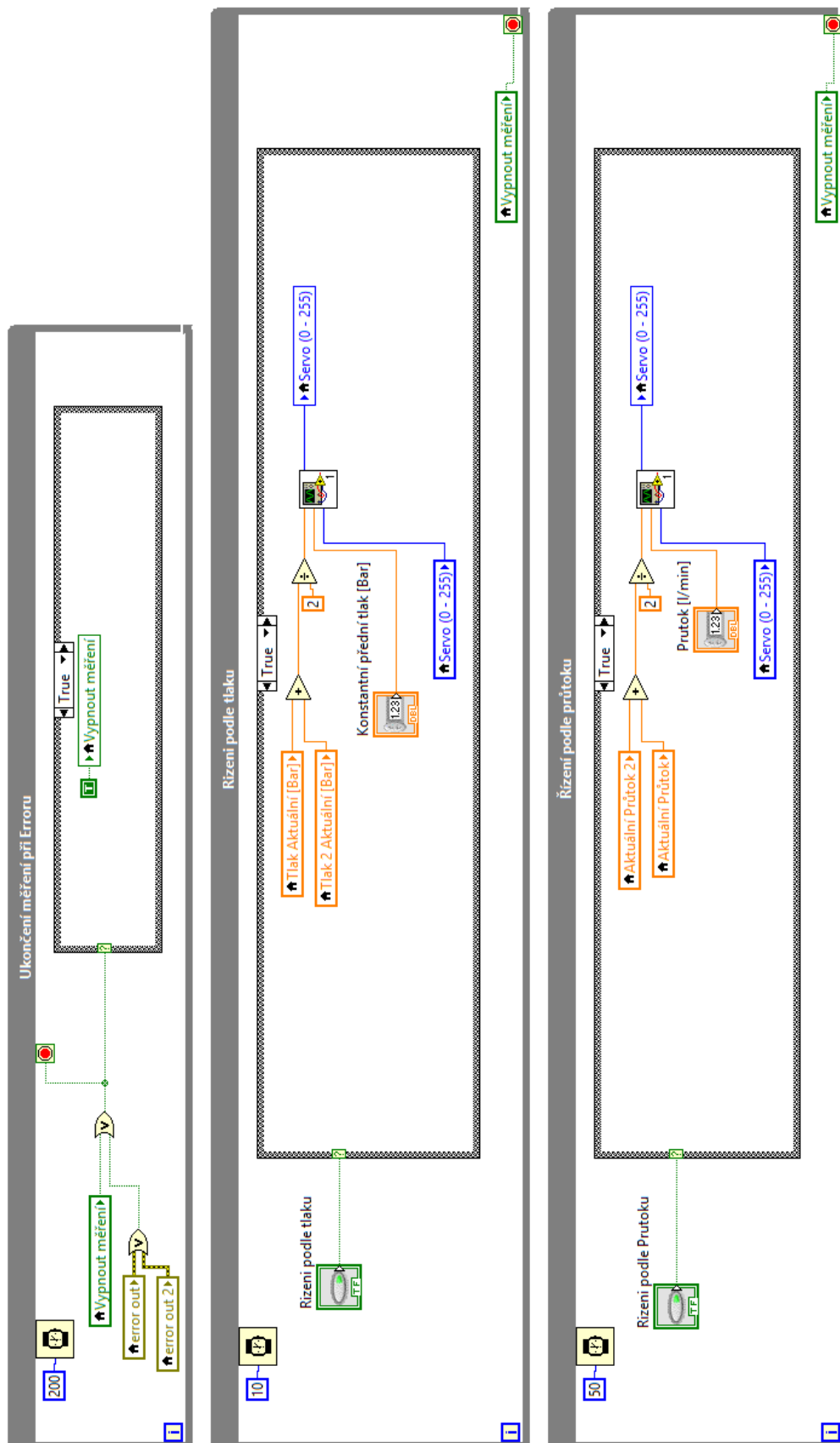
## Příloha č. 4: Druhá část programu: inicializace hodnot v Arduino



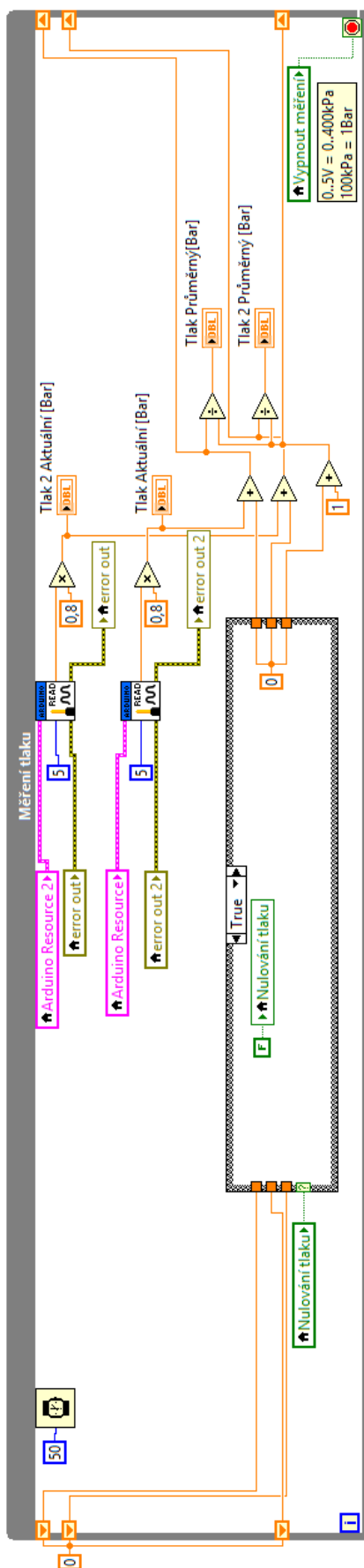
## Příloha č. 5: Vlákna pro zobrazování, nastavování servopohonu a čtení otáček



## Příloha č. 6: Vlákna ukončení při Erroru a řízení podle tlaku nebo průtoku

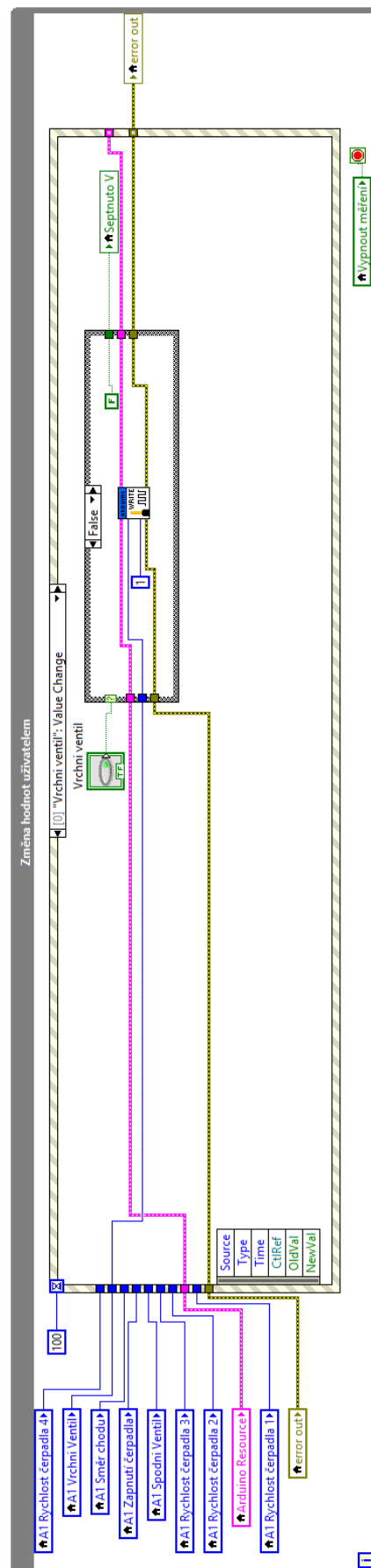


## Příloha č. 7: Vlákno měření tlaku

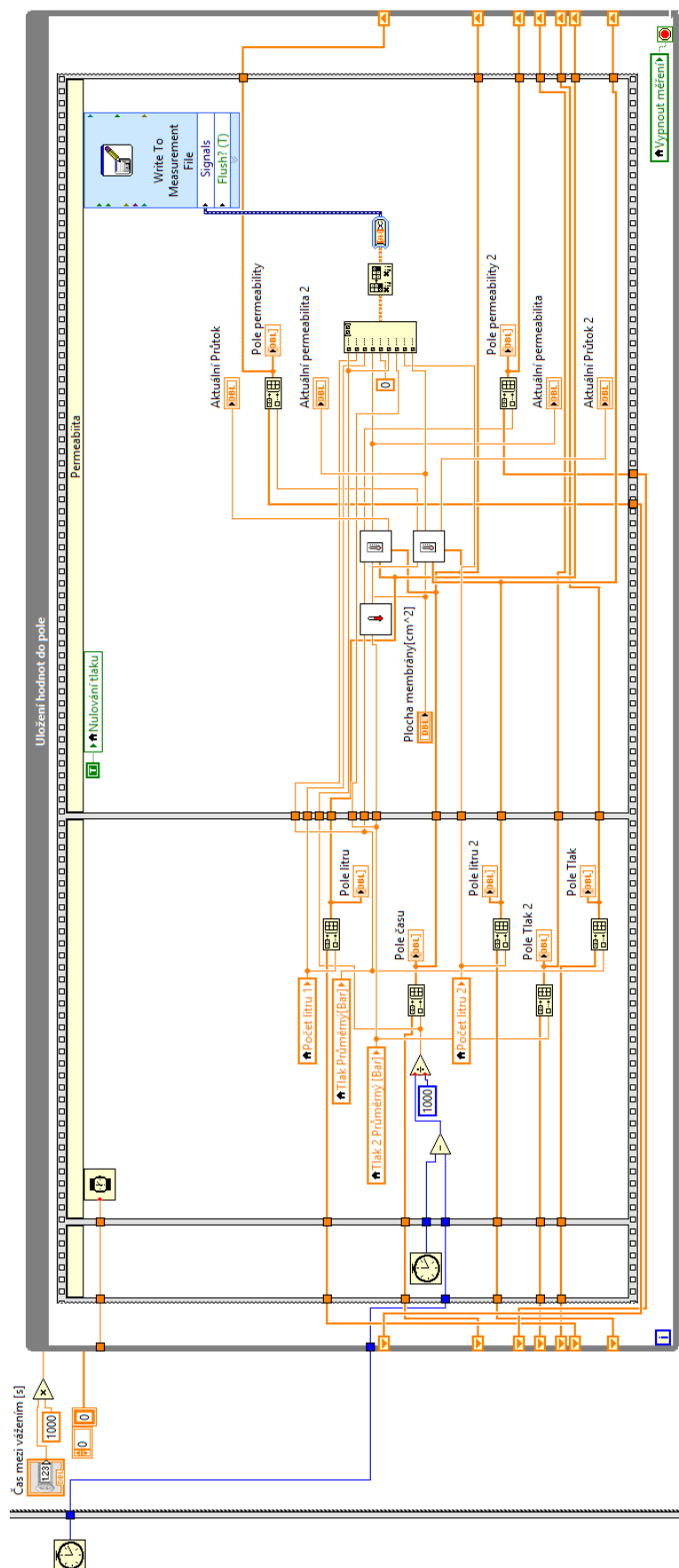




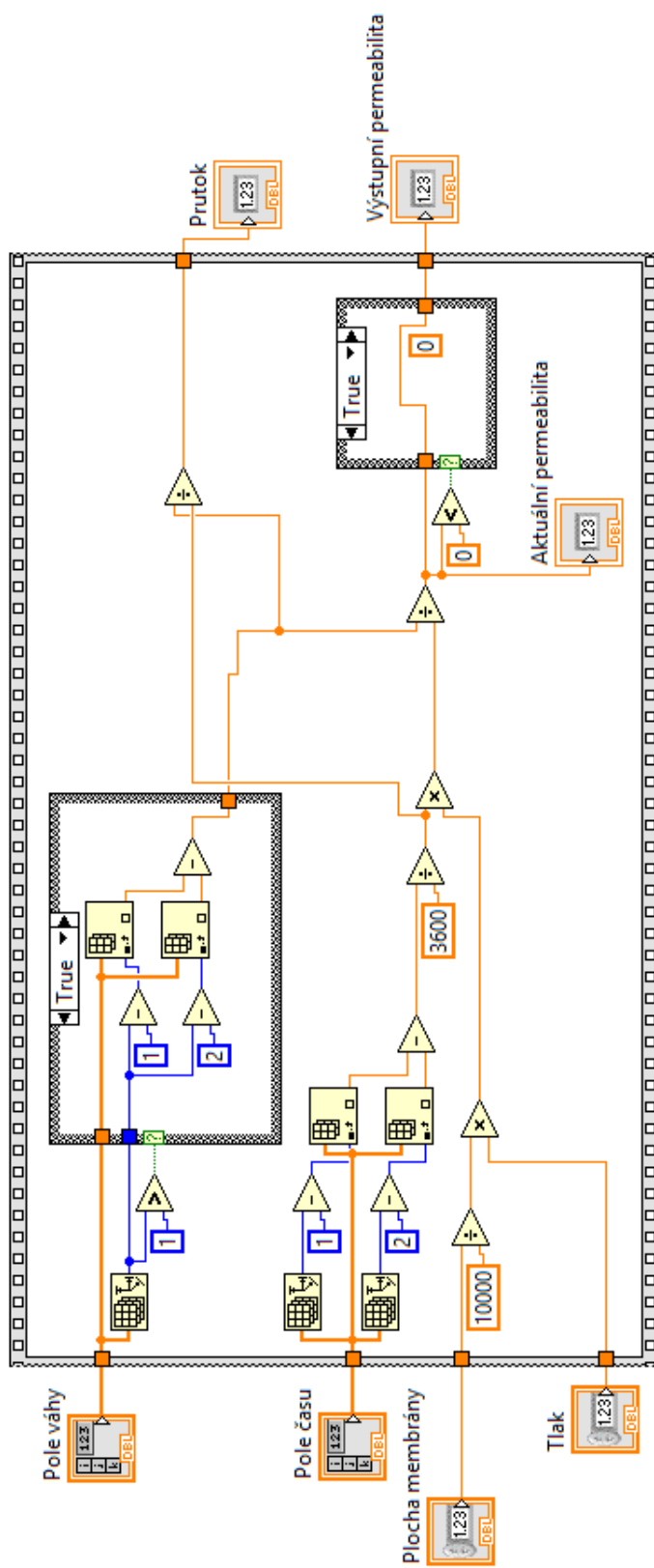
## Příloha č. 8: Vlákno pro změnu hodnot uživatelem



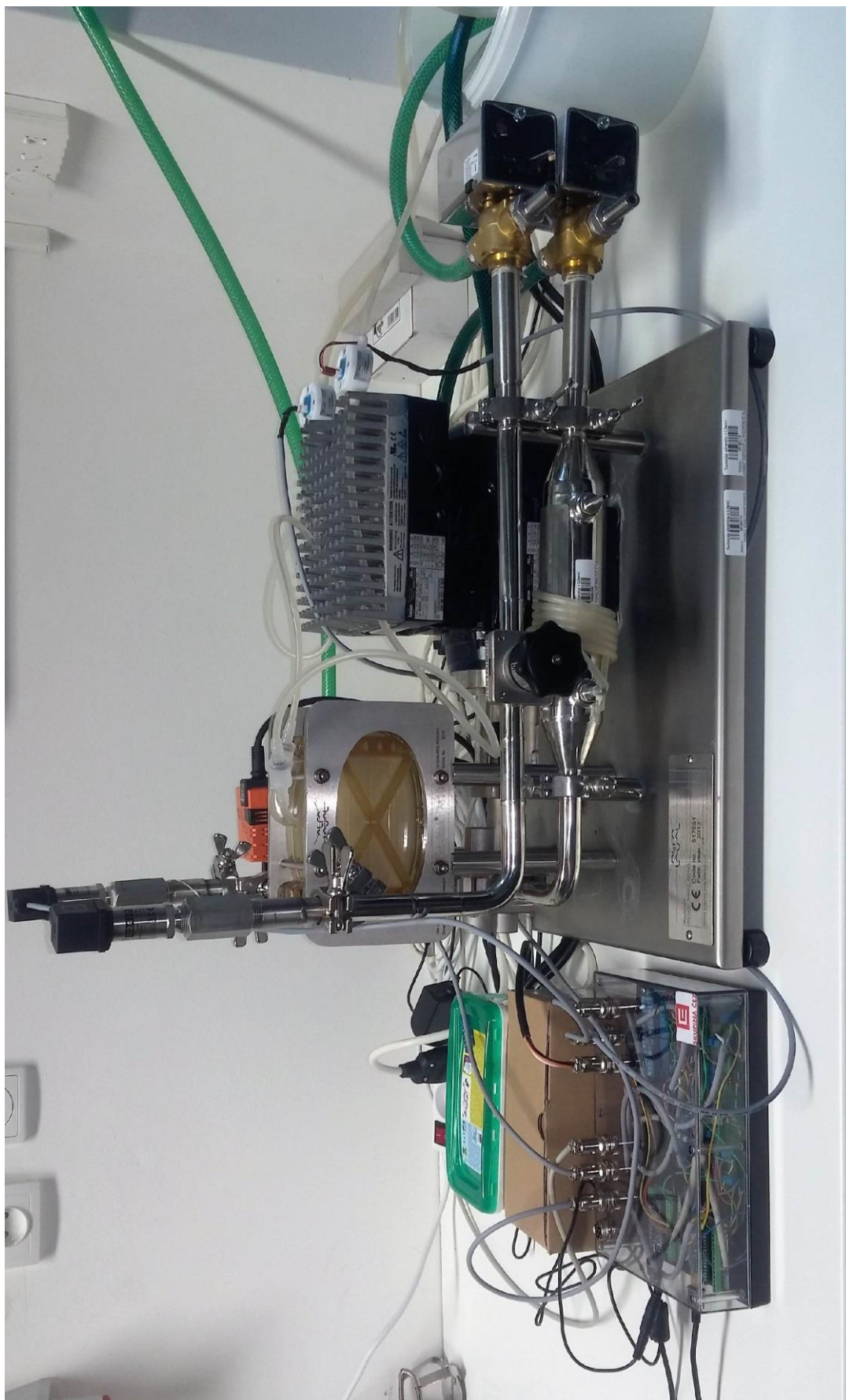
## Příloha č. 9: Vlákno pro výpočet permeability a uložení hodnot



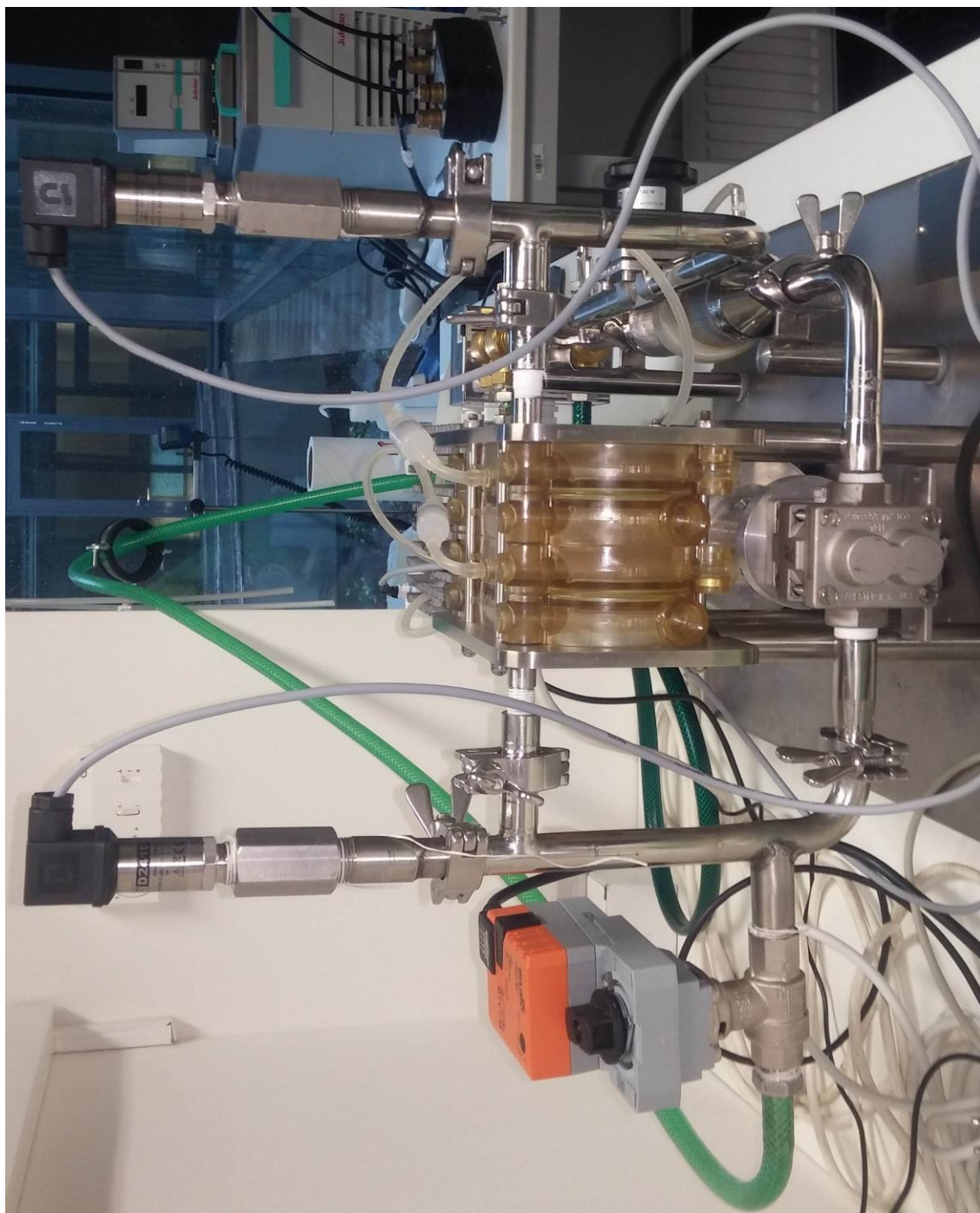
## Příloha č. 10: Programový výpočet permeability



**Příloha č. 11: Fotografie celé filtrační jednotky**

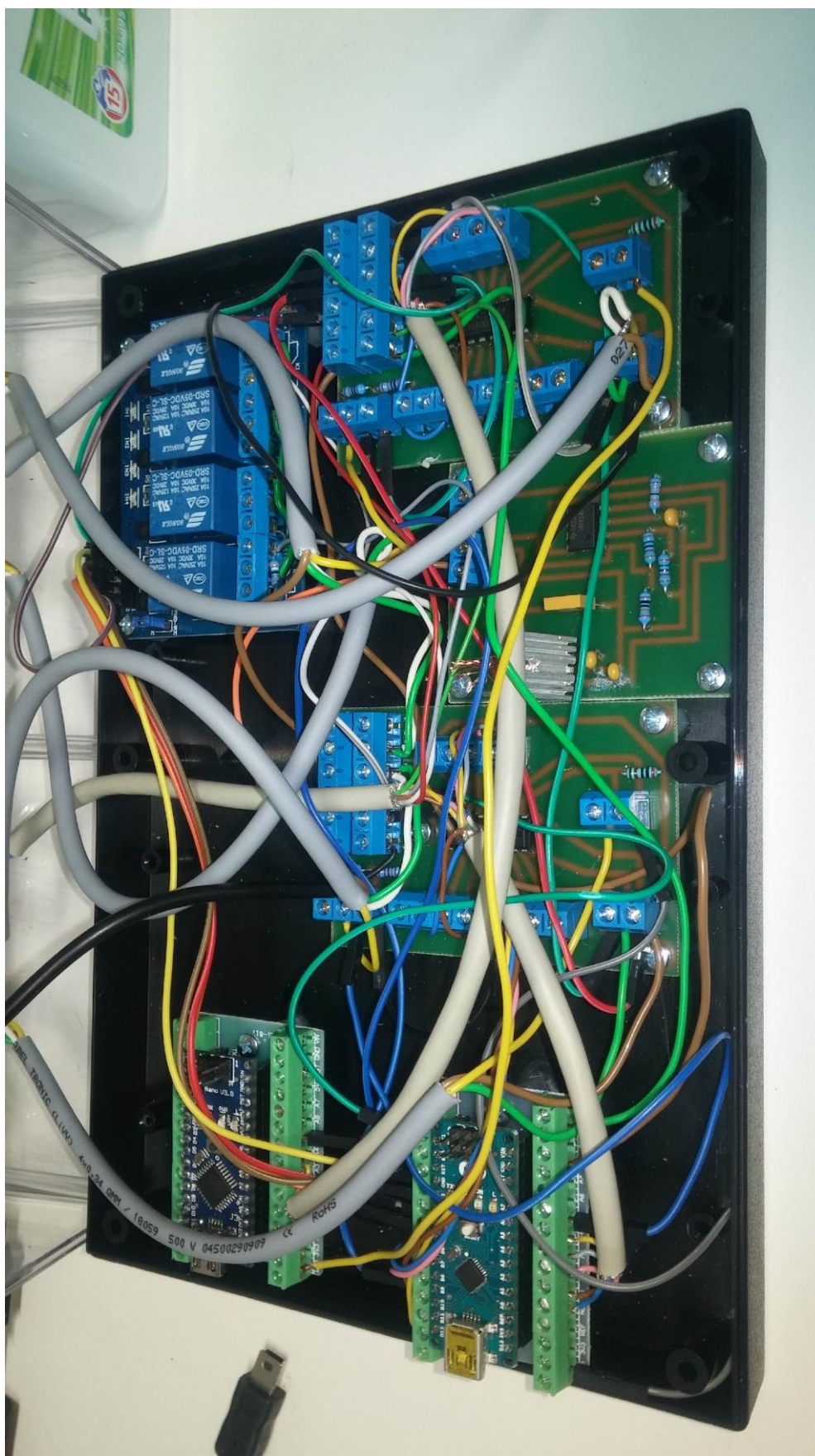


**Příloha č. 12: Fotografie filtrační jednotky z bočního pohledu**





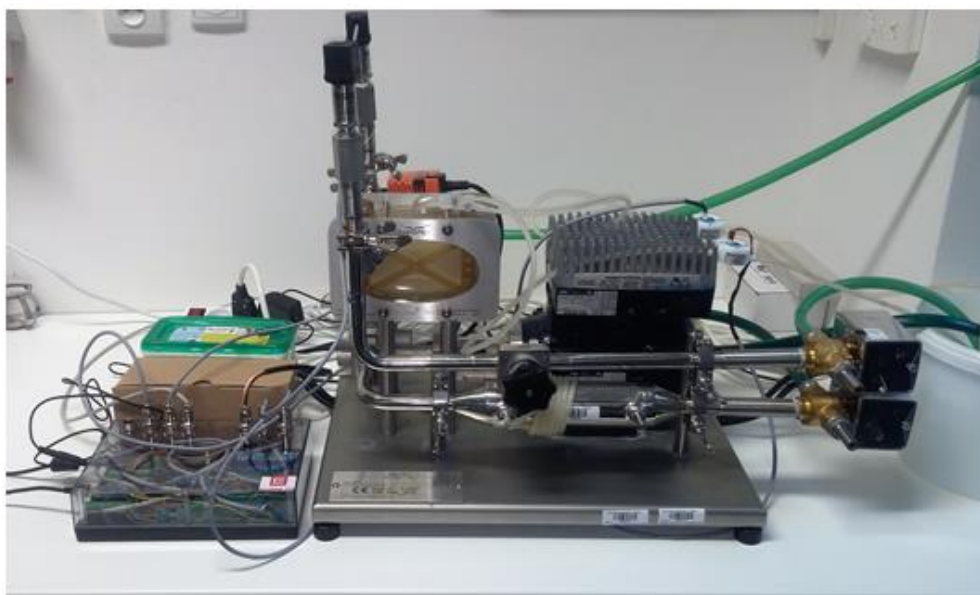
**Příloha č. 13: Fotografie ovládací elektroniky**



## Příloha č. 14: Uživatelský manuál – externí příloha

Součástí bakalářské práce je přiložené uživatelský manuál s obsahem:

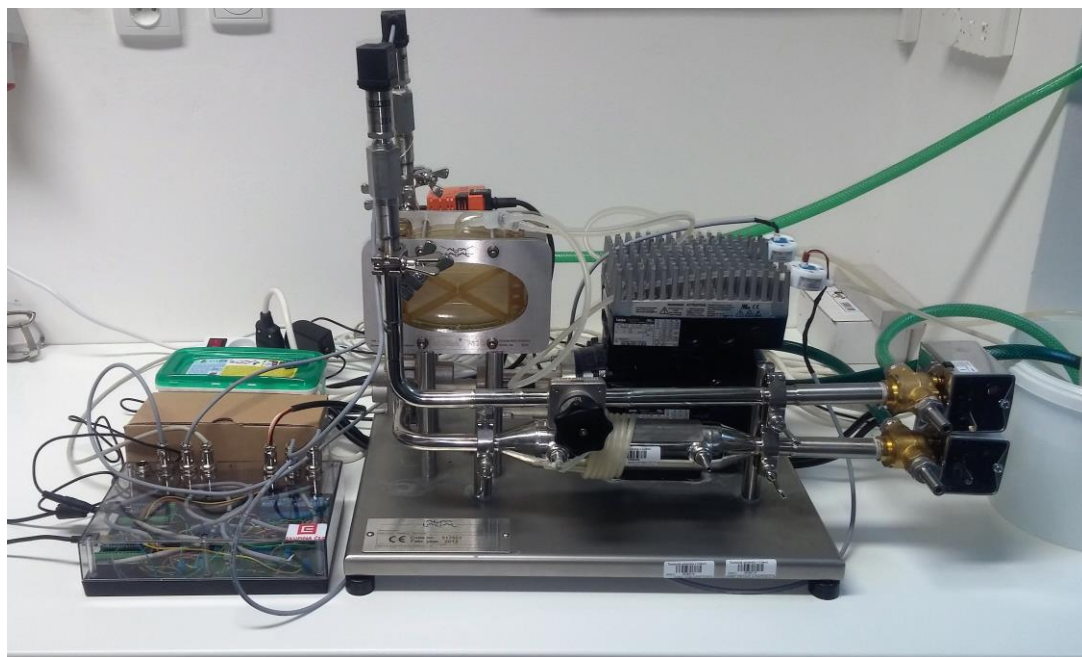
# Uživatelský manuál pro ovládání filtrační jednotky



## Obsah:

1	Hlavní zásady .....	2
2	Zapojení kabelů filtrační jednotky .....	2
3	Připojení Arduino k počítači .....	3
3.1	Ověření instalace ovladačů .....	3
3.2	Nastavení portů Arduino .....	4
4	Spuštění programu pro filtrační jednotku .....	6
5	Popis tlačítek ovládacího panelu .....	7
6	Chyba: selhání inicializace Arduino .....	8
7	Výstupní soubor .....	9

# Uživatelský manuál pro ovládání filtrační jednotky



## Obsah:

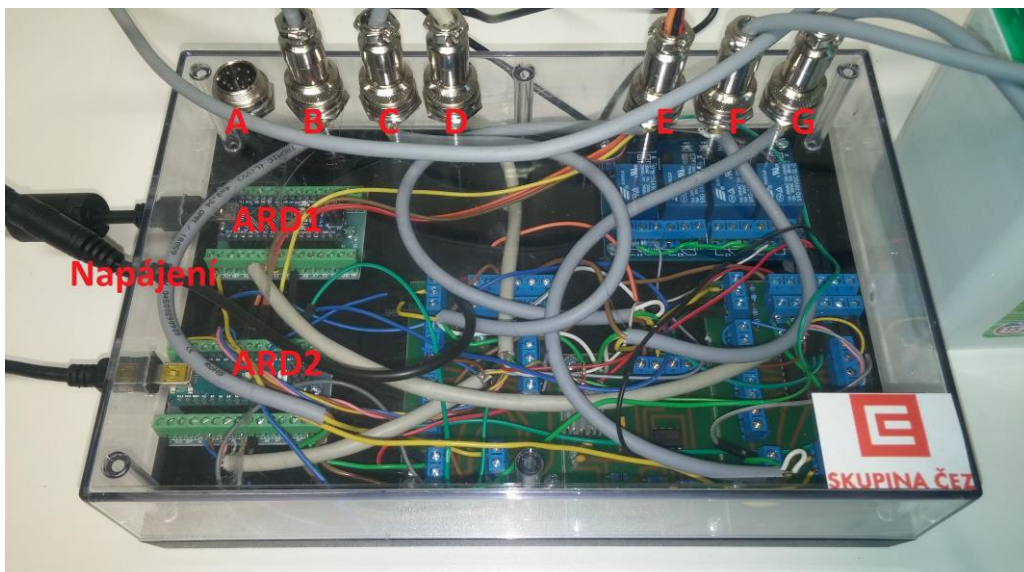
1	Hlavní zásady .....	2
2	Zapojení kabelů filtrační jednotky .....	2
3	Připojení Arduino k počítači .....	3
3.1	Ověření instalace ovladačů .....	3
3.2	Nastavení portů Arduina .....	4
4	Spuštění programu pro filtrační jednotku .....	6
5	Popis tlačítek ovládacího panelu .....	7
6	Chyba: selhání inicializace Arduina .....	8
7	Výstupní soubor .....	9



# 1 Hlavní zásady

- Před spuštěním počítače mít filtrační jednotku kompletně odpojenou od napájení;
- Napájení filtrační jednotky zapojovat až po inicializaci Arduina.

## 2 Zapojení kabelů filtrační jednotky



Obrázek č. 1: Připojení konektorů

- A – prázdný konektor
- B – zadní průtokoměr
- C – zadní tlakoměr
- D – ovládání čerpadla
- E – servopohon
- F – přední tlakoměr
- G – přední průtokoměr
- ARD1 – první datový kabel k Arduinu
- ARD2 – druhý datový kabel k Arduinu
- Napájení – připojení pro napájení 24 V

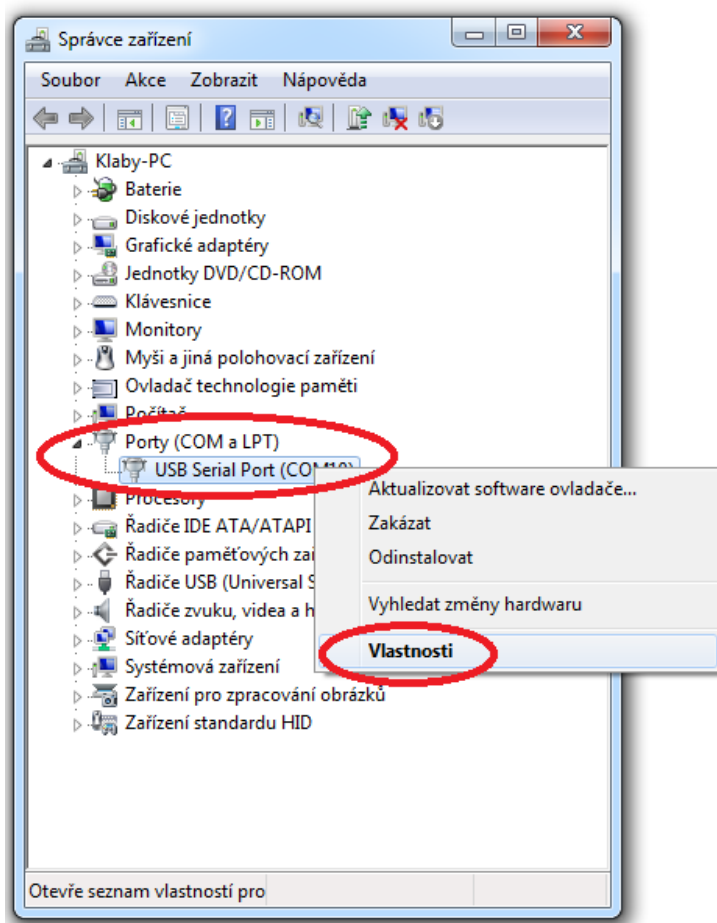
### 3 Připojení Arduino k počítači

Je vhodné připojit nejdříve jedno Arduino a nastavit podle kapitoly 3.2.

#### 3.1 Ověření instalace ovladačů

Po připojení Arduino k počítači je třeba ověřit správnost instalace ovladačů

Nainstalování ovladačů se dá ověřit ve Windows 7 pomocí „Správce zařízení“. Pokud je ovladač správně nainstalovaný, nalezneme zde řádek „Porty (COM a LPT)“, který po rozbalení obsahuje „USB Serial Port (COM?)“, kde otazník je automaticky přiřazené číslo portu. Takto nainstalovaný ovladač je znázorněn na obrázku č. 2. Pokud je však Arduino nalezeno jako neznámé zařízení, je potřeba na stránkách „<http://www.ftdichip.com/FTDrivers.htm>“ stáhnout příslušný ovladač a ten nainstalovat.

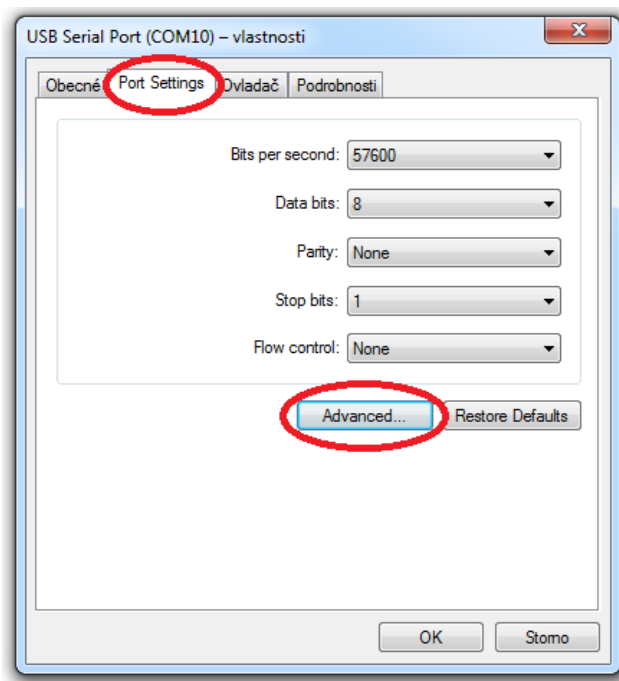


Obrázek č. 2: Zobrazení COM portu ve Správci zařízení ve Windows 7

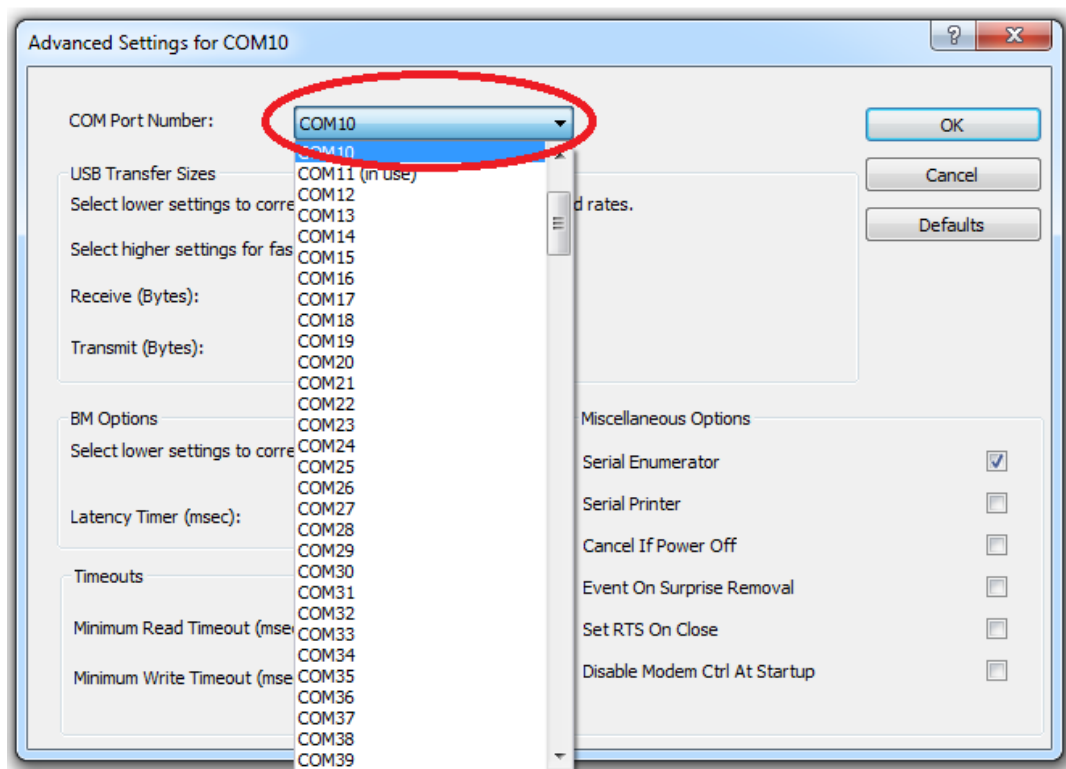
### 3.2 Nastavení portů Arduino

Po připojení desky se objeví správně nainstalované zařízení ve „Správci zařízení Windows“ jako COM port. Na toto zařízení klikneme pravým tlačítkem a zvolíme možnost vlastnosti, viz obrázek č. 2. Dále je potřeba přejít do záložky „Port Settings“ a kliknout na možnost „Advances“, viz obrázek č. 3.

Zde je nutné zvolit vlastní číslo COM portu pro dané Arduino. Tato možnost je znázorněna na obrázku č. 4. Zde je potřeba vybrat u prvního Arduino vybrat COM 9 a u druhého Arduino vybrat COM 10. Uložení volby stačí kliknout na tlačítko „OK“ a zbylá okna zavřít.



Obrázek č. 3: Vlastnoti COM portu ve Správci zařízení ve Windows 7



Obrázek č. 4: Pokročilé vlastnosti COM portu ve Správci zařízení ve Windows 7

## 4 Spuštění programu pro filtrační jednotku

- Po správném spuštění bez chyb je potřeba nastavit počet membrán, které jsou použité ve filtrační jednotce, a čas, po kterém se budou ukládat hodnoty měření. V základním nastavení je čas vážení 1 vteřina a počet membrán 4.
- Připojit napájení filtrační jednotky.
- Dále stačí kliknout na tlačítko „Spustit měření“. Toto nastavení a spuštění je ve vrchní části ovládacího panelu, který je zobrazen na obrázku č. 5.



Obrázek č. 5: Ovládací panel programu filtrační jednotky

## 5 Popis tlačítek ovládacího panelu

- Vrchní ventil

Přepnutí vrchního třícestného ventilu pro vstup/výstup jiné kapaliny.

- Spodní ventil

Přepnutí spodního třícestného ventilu pro vstup/výstup jiné kapaliny.

- Směr chodu

Při sepnutí je určen chod čerpadla vpřed. Při vypnutí tlačítka je zapnutý zpětný chod čerpadla.

- Zapnutí čerpadla

Zapíná čerpadlo.

- Spustit měření

Tlačítko pro spuštění měření a zaznamenávání hodnot.

- Řízení podle tlaku

Při stisknutí tohoto tlačítka se systém bude regulovat na hodnotu tlaku, který je napsán v kolonce „Konstantní přední tlak“

- Řízení podle průtoku

Při stisknutí tohoto tlačítka se systém bude regulovat na hodnotu průtoku, který je napsán v kolonce „Konstantní přední tlak“

- Rychlost čerpadla

Otočné tlačítko pro volbu rychlosti čerpadla s devíti stupni.

## 6 Chyba: selhání inicializace Arduina

Při zahájení programu je možné zobrazení hlášky, která je zobrazena na obrázku č. 6. Tato hláška se může zobrazit ze tří důvodů.

### 1) Nepřipojené Arduino

Zkontrolovat připojení USB kabelů od Arduina k počítači a opakovat. \*

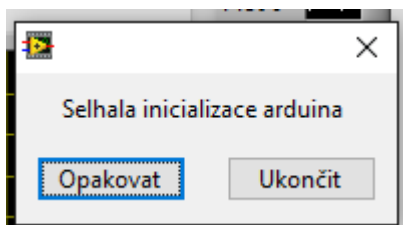
### 2) Špatné nastavení COM portů

Zopakovat nastavení v kapitole 3 a opakovat. \*

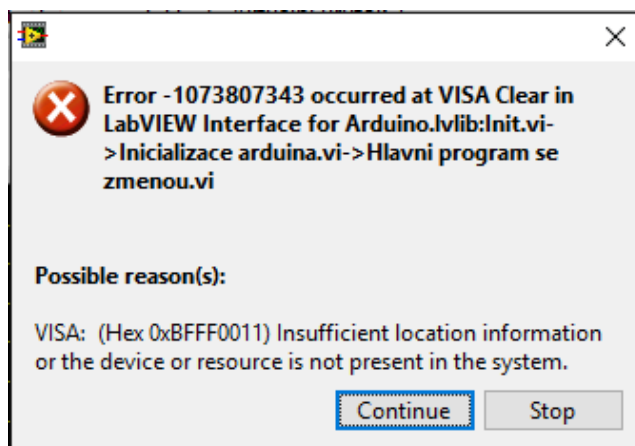
### 3) Špatné ukončení Arduino

Kliknout na možnost opakovat. \*

\* Pomocí opakování inicializace Arduina se vymaže paměť Arduina a opět se pokusí připojit. Díky tomuto postupu se zobrazí výstupní hláška Arduina, která je zobrazena na obrázku č. 7. Zde je potřeba kliknout na možnost „Continue“. Tato hláška se zobrazí dvakrát.



Obrázek č. 6: Selhání inicializace



Obrázek č. 7: Ukončování Arduina

## 7 Výstupní soubor

Soubor s údaji o měření nalezneme ve složce „C:\Mereni\“ pod názvem „Mereni.xlsx“. Tento soubor obsahuje tabulku s osmi potřebnými údaji a je zobrazena na obrázku č. 8.

Popis sloupců:

- A) Čas měření
- B) Tlak zadního tlakoměru
- C) Protečené množství zadním průtokoměrem
- D) Vypočtená permeabilita zadní dvojice membrán
- E) Oddělovací sloupec
- F) Tlak předního tlakoměru
- G) Protečené množství předním průtokoměrem
- H) Vypočtená permeabilita přední dvojice membrán

	A	B	C	D	E	F	G	H	I
1	Untitled	Untitled 1	Untitled 2	Untitled 3	Untitled 4	Untitled 5	Untitled 6	Untitled 7	Untitled 8
2	0,027	1,5994	0,102674	273256,1	0	0,027	1,49104	0,029947	74300
3	30,109	1,52096	0,038503	-179,573	0	30,109	1,272784	0,008556	-50,0907
4	60,231	1,5288	0,085561	117,3095	0	60,231	1,42688	0,025668	39,81412
5	90,362	1,54448	0,13262	116,6336	0	90,362	1,43472	0,038503	29,54861
6	120,471	1,54056	0,179679	117,0386	0	120,471	1,4308	0,051337	29,64544
7	150,654	1,49744	0,226738	119,7032	0	150,654	1,39552	0,064171	30,42433
8	180,784	1,20736	0,265241	121,699	0	180,784	1,12504	0,077005	37,80045
9	210,927	1,19952	0,299465	108,5083	0	210,927	1,12112	0,085561	25,35407
10	241,067	1,2152	0,337968	121,2362	0	241,067	1,12896	0,094118	25,02941
11	271,228	1,29752	0,376471	114,0252	0	271,228	1,19952	0,102674	23,42512
12	301,328	1,29752	0,414973	113,8841	0	301,328	1,20344	0,11123	23,47259
13	331,457	1,20148	0,449198	110,0986	0	331,457	1,10544	0,119786	25,32447
14	361,651	1,196907	0,483422	108,8962	0	361,651	1,11524	0,128342	25,36651
15	391,794	1,19952	0,521925	122,9315	0	391,794	1,11328	0,136898	25,35407
16	421,874	1,23872	0,55615	105,4182	0	421,874	1,1564	0,145455	24,60315
17	451,944	1,245113	0,594652	118,89	0	451,944	1,153919	0,154011	24,48495
18	482	1,24552	0,628877	105,7153	0	482	1,154071	0,162567	24,48837
19	512,057	1,246606	0,66738	118,8001	0	512,057	1,155292	0,166845	12,23311
20	542,107	1,24656	0,701604	105,5235	0	542,107	1,1564	0,175401	24,47282
21	572,262	1,24656	0,740107	118,3005	0	572,262	1,1564	0,183957	24,3876

Obrázek č. 8: Výstupní tabulka v souboru „Mereni.xlsx“